

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference PC/MPF/21646	FOR FURTHER ACTION see Notification of Transmittal of International Search Report (Form PCT/ISA/220) as well as, where applicable, item 5 below.	
International application No. PCT/IB 00/01486	International filing date (day/month/year) 26/09/2000	(Earliest) Priority Date (day/month/year) 27/09/1999
Applicant CANAL+ SOCIETE ANONYME et al.		

This International Search Report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This International Search Report consists of a total of 3 sheets.

☒ It is also accompanied by a copy of each prior art document cited in this report.

1. Basis of the report

- a. With regard to the **language**, the international search was carried out on the basis of the international application in the language in which it was filed, unless otherwise indicated under this item.

☐ the international search was carried out on the basis of a translation of the international application furnished to this Authority (Rule 23.1(b)).

- b. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international search was carried out on the basis of the sequence listing :

☐ contained in the international application in written form.

☐ filed together with the international application in computer readable form.

☐ furnished subsequently to this Authority in written form.

☐ furnished subsequently to this Authority in computer readable form.

☐ the statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.

☐ the statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished

2. ☐ **Certain claims were found unsearchable** (See Box I).

3. ☐ **Unity of invention is lacking** (see Box II).

4. With regard to the **title**,

☒ the text is approved as submitted by the applicant.

☐ the text has been established by this Authority to read as follows:

5. With regard to the **abstract**,

☒ the text is approved as submitted by the applicant.

☐ the text has been established, according to Rule 38.2(b), by this Authority as it appears in Box III. The applicant may, within one month from the date of mailing of this international search report, submit comments to this Authority.

6. The figure of the **drawings** to be published with the abstract is Figure No.

☒ as suggested by the applicant.

☐ because the applicant failed to suggest a figure.

☐ because this figure better characterizes the invention.

Zb _____

☐ None of the figures.

INTERNATIONAL SEARCH REPORT

International Application No

PCT/IB 00/01486

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/44

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 831 615 A (KOGAN DAN D ET AL) 3 November 1998 (1998-11-03) abstract -/--	1-8, 10-12, 14-18, 20-25, 29-34, 39, 45-47, 49-51, 53-57, 59-64, 68-73, 80, 86-88, 90-92, 94-98, 100-105, 109-114, 118

☒ Further documents are listed in the continuation of box C.☒ Patent family members are listed in annex.

* Special categories of cited documents :

A document defining the general state of the art which is not considered to be of particular relevance

E earlier document but published on or after the international filing date

L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

O document referring to an oral disclosure, use, exhibition or other means

P document published prior to the international filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

X document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

Y document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

& document member of the same patent family

Date of the actual completion of the international search:

17 April 2001

Date of mailing of the international search report

25/04/2001

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Fonderson, A

INTERNATIONAL SEARCH REPORT

International Application No

PCT/IB 00/01486

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	column 8, line 17 - line 58 column 11, line 32 -column 12, line 28 figure 13 --- EP 0 806 756 A (IBM) 12 November 1997 (1997-11-12)	1,2,6,7, 10,11, 16-18, 20,31, 39-41, 45,46, 49,50, 55-57, 59,70, 80-82, 86,87, 90,91, 96-98, 100,111, 118
A	the whole document --- WO 99 17549 A (SUN MICROSYSTEMS INC) 8 April 1999 (1999-04-08) page 10, paragraph 3 -page 11, paragraph 2 -----	1-125

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/IB 00/01486

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5831615	A	03-11-1998	US	5768607 A	16-06-1998
			US	5893126 A	06-04-1999
EP 0806756	A	12-11-1997	US	5764229 A	09-06-1998
			JP	10039850 A	13-02-1998
WO 9917549	A	08-04-1999	AU	9476898 A	23-04-1999
			EP	1018268 A	12-07-2000

17 DEC 2001

PCT

WIPO

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference PC/MGM/21646	FOR FURTHER ACTION See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)	
International application No. PCT/IB00/01486	International filing date (day/month/year) 26/09/2000	Priority date (day/month/year) 27/09/1999
International Patent Classification (IPC) or national classification and IPC G06F9/44		
Applicant CANAL+ SOCIETE ANONYME et al.		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.



2. This REPORT consists of a total of 7 sheets, including this cover sheet.

- ☐ This report is also accompanied by ANNEXES, i.e. sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

These annexes consist of a total of sheets.

3. This report contains indications relating to the following items:

- I ☒ Basis of the report
- II ☐ Priority
- III ☒ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
- IV ☐ Lack of unity of invention
- V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement
- VI ☐ Certain documents cited
- VII ☒ Certain defects in the international application
- VIII ☒ Certain observations on the international application

Date of submission of the demand 27/04/2001	Date of completion of this report 13.12.2001
Name and mailing address of the international preliminary examining authority:  European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Authorized officer Lacroix, P Telephone No. +49 89 2399 2707 

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No. PCT/IB00/01486

I. Basis of the report

1. With regard to the **elements** of the international application (*Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report since they do not contain amendments (Rules 70.16 and 70.17)*):
Description, pages:

1-36 as originally filed

Claims, No.:

1-125 as originally filed

Drawings, sheets:

1-16 as originally filed

2. With regard to the **language**, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.

These elements were available or furnished to this Authority in the following language: , which is:

- ☐ the language of a translation furnished for the purposes of the international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of a translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

4. The amendments have resulted in the cancellation of:

- ☐ the description, pages:
- ☐ the claims, Nos.:

**INTERNATIONAL PRELIMINARY
EXAMINATION REPORT**

International application No. PCT/IB00/01486

☐ the drawings, sheets:

5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed (Rule 70.2(c)):

(Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.)

6. Additional observations, if necessary:

III. Non-establishment of opinion with regard to novelty, inventive step and industrial applicability

1. The questions whether the claimed invention appears to be novel, to involve an inventive step (to be non-obvious), or to be industrially applicable have not been examined in respect of:

☐ the entire international application.

☒ claims Nos. 124,125.

because:

☐ the said international application, or the said claims Nos. relate to the following subject matter which does not require an international preliminary examination (*specify*):

☒ the description, claims or drawings (*indicate particular elements below*) or said claims Nos. 124,125 are so unclear that no meaningful opinion could be formed (*specify*):
see separate sheet

☐ the claims, or said claims Nos. are so inadequately supported by the description that no meaningful opinion could be formed.

☐ no international search report has been established for the said claims Nos. .

2. A meaningful international preliminary examination cannot be carried out due to the failure of the nucleotide and/or amino acid sequence listing to comply with the standard provided for in Annex C of the Administrative Instructions:

☐ the written form has not been furnished or does not comply with the standard.

☐ the computer readable form has not been furnished or does not comply with the standard.

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)

Yes: Claims 1-123

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No. PCT/IB00/01486

	No:	Claims	
Inventive step (IS)	Yes:	Claims	9,13,19,26-28,35-38,40-44,48,52,58,65-67,74-79,81-85,89,93,99,106-108,115-117
	No:	Claims	1-8,10-12,14-18,20-25,29-34,39,45-47,49-51,53-57,59-64,68-73,80,86-88,90-92,94-98,100-105,109-114,118-123
Industrial applicability (IA)	Yes:	Claims	1-123
	No:	Claims	

2. Citations and explanations
see separate sheet

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:
see separate sheet

VIII. Certain observations on the international application

The following observations on the clarity of the claims, description, and drawings or on the question whether the claims are fully supported by the description, are made:
see separate sheet

Re Item III

Non-establishment of opinion with regard to novelty, inventive step and industrial applicability

Claims 124, 125 do not require an international preliminary examination since they are so unclear that no meaningful opinion could be formed as they rely exclusively on references to the drawings. Cf. Rule 6.2 (a) PCT and Preliminary Examination Guidelines PCT S-07/1998, Page 20, Section III, 4.10.

Re Item V

Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

- 1). Reference is made to the following documents:

D1: WO 99 17549 A (SUN MICROSYSTEMS INC) 8 April 1999 (1999-04-08)

D2: US-A-5 831 615 (KOGAN DAN D ET AL) 3 November 1998 (1998-11-03)

- 2). Document D1 discloses an apparatus and a method for displaying a window (browser panel) on a screen wherein the window has a background (background of the browser panel and its content) through which underlying objects are visible (since the browser panel may be made partially transparent - Cf. Page 10), the method being carried out by a receiver/decoder (Box 2 receiving and decoding television signals - Cf. Page 7), the receiver/decoder being used with a television set (Cf. Page 6).

Document D1 provides further a strong indication of using transparent windows such that the person skilled in the art would obviously consult prior art documents being concerned with transparent windows such as D2. The person skilled in the art will then be able to combine the teachings of D1 and D2 without exercising inventive skill.

As can be seen from the drawings of D2, it appears that the method of displaying a window as disclosed at Column 11, Lines 30-58, comprises the steps of:

defining the size of the window (frame);

drawing the foregrounds objects in the window; and
arranging the background of the window such that objects underlying the
background are visible (Fig. 10 a).

The drawing of the window can be corrected (Cf. D2, Column 8, Lines 59-63).
Due to the OLE properties, it appears obvious to implement first and second
modes of operation with appropriate annotations (Cf. D2, Column 10, Lines 14-
59).

- 3). The features of the objected dependent claims are either known from D1-D2
or form part of the general knowledge of the person skilled in the art.
They do not appear to comprise anything which would go beyond the prior art to
an extent that it could be considered as involving an inventive step.

Re Item VII

Certain defects in the international application

- 1). Contrary to the requirements of Rule 5.1(a)(ii) PCT, the relevant background
art disclosed in the documents D1, D2 is not mentioned in the description, nor are
these documents identified therein.
- 2). The independent claims are not in the two-part form in accordance with Rule
6.3(b) PCT, which in the present case would be appropriate since D1 and/or D2
disclose to a large extent the most of the features implementing the core of the
current application.
The independent claims should therefore be redrafted accordingly.
- 3). The features of the claims are not provided with reference signs placed in
parentheses (Rule 6.2(b) PCT).

Re Item VIII

Certain observations on the international application

Although a plurality of claims in the same category have been drafted as
separate independent claims, they appear to relate effectively to the same
subject-matter and to differ from each other only with regard to the definition of the
subject-matter for which protection is sought ..and/or.. in respect of the
terminology used for the features of that subject-matter. The aforementioned

**INTERNATIONAL PRELIMINARY
EXAMINATION REPORT - SEPARATE SHEET**

International application No. PCT/IB00/01486

claims therefore lack conciseness. Moreover, lack of clarity of the claims as a whole arises, since the plurality of independent claims makes it difficult, if not impossible, to determine the matter for which protection is sought.

Hence, said claims do not meet the requirements of Article 6 PCT.

In order to overcome this objection, it would appear appropriate to file an amended set of claims defining the relevant subject-matter in terms of /a single/a minimum number of/ independent claim/claims in each category followed by dependent claims covering features which are merely optional (Rule 6.4 PCT).



DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. CL7)
X	EP 0 806 756 A (IBM) 12 November 1997 (1997-11-12) * the whole document *	1-8, 13, 18-20	G06F3/033 G09G5/14
X	EP 0 605 945 A (FIRSTPERSON INC) 13 July 1994 (1994-07-13) * abstract; claims; figures *	1-6, 13, 18-20	
A	EP 0 670 652 A (FIRSTPERSON INC) 6 September 1995 (1995-09-06) * abstract; figures *	17	
X	US 5 831 615 A (KOGAN DAN D ET AL) 3 November 1998 (1998-11-03) * column 12, line 10 - line 41; figure 13 *	1-16, 18-22	
X	WO 98 09270 A (CASE STEVEN V ;VIA INC (US); CARROLL DAVID W (US); CARROLL JAMES L) 5 March 1998 (1998-03-05) * page 13, line 21 - page 14, line 9; claims 26,27; figures 4-6 *	1-7, 9-11, 13-16, 18-22	TECHNICAL FIELDS SEARCHED (Int. CL7) G06F G09G
The present search report has been drawn up for all claims			
Place of search BERLIN		Date of completion of the search 11 April 2000	Examiner Durand, J
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			



CLAIMS INCURRING FEES

The present European patent application comprised at the time of filing more than ten claims.

☐ Only part of the claims have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims and for those claims for which claims fees have been paid, namely claim(s):

☐ No claims fees have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims.

LACK OF UNITY OF INVENTION

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

see sheet B

☒ All further search fees have been paid within the fixed time limit. The present European search report has been drawn up for all claims.

☐ As all searchable claims could be searched without effort justifying an additional fee, the Search Division did not invite payment of any additional fee.

☐ Only part of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the inventions in respect of which search fees have been paid, namely claims:

☐ None of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the invention first mentioned in the claims, namely claims:



European Patent
Office

**LACK OF UNITY OF INVENTION
SHEET B**

Application Number
EP 99 40 2353

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

1. Claims: 1-8 13 17-20

drawing in and displaying overlapping windows

2. Claims: 9-12 14-16 21 22

drawing in a window with two modes of affecting an other window

ANNEX THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO.

EP 99 40 2353

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

11-04-2000

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0806756 A	12-11-1997	US 5764229 A JP 10039850 A	09-06-1998 13-02-1998
EP 0605945 A	13-07-1994	DE 69315969 D DE 69315969 T JP 6301505 A US 5651107 A US 5999191 A	05-02-1998 30-07-1998 28-10-1994 22-07-1997 07-12-1999
EP 0670652 A	06-09-1995	US 5524195 A JP 8101756 A US 5745710 A	04-06-1996 16-04-1996 28-04-1998
US 5831615 A	03-11-1998	US 5768607 A US 5893126 A	16-06-1998 06-04-1999
WO 9809270 A	05-03-1998	AU 4327597 A EP 0922275 A	19-03-1998 16-06-1999

(19) World Intellectual Property Organization
International Bureau



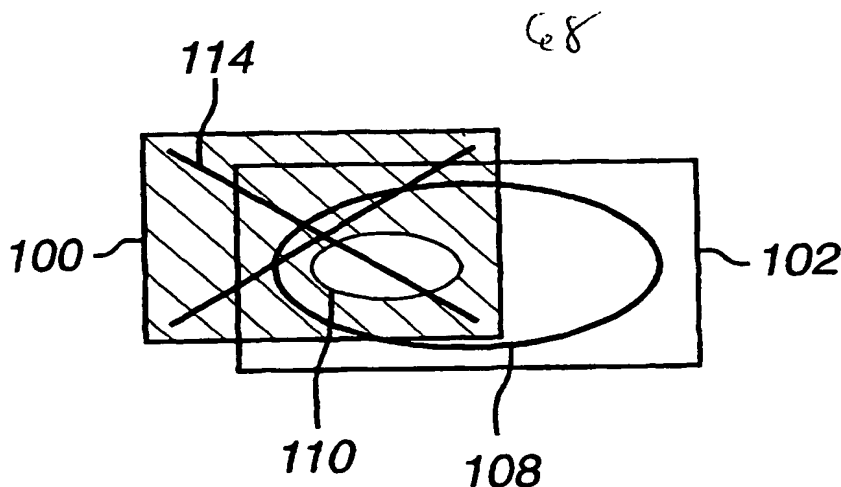
(43) International Publication Date
5 April 2001 (05.04.2001)

PCT

(10) International Publication Number
WO 01/23995 A2

- (51) International Patent Classification⁷: G06F 9/44 (74) Agents: COZENS, Paul, Dennis et al.; Mathys & Squire, 100 Gray's Inn Road, London WC1X 8AL (GB).
- (21) International Application Number: PCT/IB00/01486
- (22) International Filing Date:
26 September 2000 (26.09.2000)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
99402353.9 27 September 1999 (27.09.1999) EP
99402721.7 2 November 1999 (02.11.1999) EP
00300832.3 3 February 2000 (03.02.2000) EP
- (81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.
- (84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).
- (71) Applicant (*for all designated States except US*): CANAL+ SOCIETE ANONYME [FR/FR]; 85/89 Quai Andre Citroen, F-75711 Paris Cedex 15 (FR).
- Published:
— Without international search report and to be republished upon receipt of that report.
- (72) Inventor; and
- (75) Inventor/Applicant (*for US only*): RUELLE, Alex [FR/FR]; Canal+ Technologies Societe Anonyme, 34, place Raoul Dautry, F-75516 Paris Cedex 15 (FR).
- For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: WINDOWING SYSTEMS



(57) Abstract: A method of displaying a window on a screen is described, in which the window has a background through which underlying objects are visible. Also described are methods of managing windows when drawing takes place in a window which lies under such a window.

WO 01/23995 A2

WINDOWING SYSTEMS

The invention relates to apparatus for and methods of displaying a window, drawing in a window and displaying a navigator display, a receiver/decoder, a television system, a
5 computer program product, a computer readable medium and a signal.

The invention has particular application to a windowing system for use with receiver/decoders.

10 Reference is made to European Patent Applications Nos. 99402353.9, 99402721.7 and 00300832.3, whose content is herein incorporated by reference, and from which Convention priority is claimed.

Windowing systems are used to allow different window contents, for example graphical
15 objects such as text or graphical images, to be displayed simultaneously in different areas, known as windows, on a screen. Windows in conventional windowing systems comprise rectangles in a area of which graphical objects may be drawn for display against an opaque background. Windows generally occupy a selected area of the screen, a size, a background colour and a position of a window normally being defined on opening the
20 window and alterable after opening.

Conventional windowing systems may be inflexible in the way they allow information to be presented on the screen. For example, if two windows overlap, then the foremost window will obscure the overlapped part of the underlying window.

25

In a first aspect of the present invention there is provided a method of displaying a window on a screen wherein the window has a background through which underlying objects are visible. In one particularly preferred aspect this method is carried out by a receiver/decoder, the receiver/decoder being preferably for use with a television set.

30

The invention can provide the advantage that information may be presented on a screen with some or all of the benefits of a windowing system, whilst objects underlying the window may be at least partially seen through the window.

- 2 -

The invention stems from the realisation that windows may have a transparent background, provided certain precautions are taken. Such precautions include not drawing behind the window once the window has been drawn, or monitoring drawing behind the window to correct for any effects of the drawing.

5

The method may further comprise drawing a frame of the window, such that the limits of the window are visible. The size of the window may be independent of the size of the screen, and the window may be resizeable and moveable on the screen.

- 10 Preferably, the window forms part of a screen comprising a window display and the method further comprises combining the window display with a video image. The window display may be combined with the video image by blending the two, but may be simply overwritten on the (preferably larger area) video image, both preferably using hardware means, obscuring the video image behind the window display. Partly resulting
- 15 from the increased compactness offered by transparent windows, this can minimise the intrusion of menus and the like on a video image, which can be particularly advantageous in the context of receiver/decoders.

Preferably, the method further comprises displaying the window on a television screen.

- 20 The present invention is particularly applicable to television displays, by comparison with blending methods, which are typically lower in contrast.

Preferably, the method further comprises the steps of:

- 25 defining the size of the window;
 drawing foreground objects in the window; and
 arranging the background of the window.

- 30 This feature is also provided independently. Hence the invention also provides a method of displaying a window comprising the steps of defining the size of the window, drawing foreground objects in the window, and arranging the background of the window such that objects underlying the background are visible. The background is preferably those parts of the window in which no foreground objects are drawn.

The step of drawing foreground objects may comprise setting foreground pixels to

- 3 -

desired values. The foreground objects may comprise any or all of a straight line, curved line, box, circle, triangle and typographical character, and preferably are drawn in at least two draw operations. The foreground objects may comprise interactive controls. Controls may be for example clickable buttons, radio buttons, text boxes, clickable arrows, and so on. This can allow floating panels of arbitrary shape to be created.

The step of arranging the background may comprise leaving at least one pixel value unaltered, or blending at least one pixel value with a pixel value of an underlying image, preferably in a region defining the background. In this way, underlying objects may be visible through the background, by making the background either partially or completely transparent.

In one embodiment, the background is completely transparent, and no background pixel values are drawn. In another embodiment, the background is assigned a transparency factor, and at least one background pixel value is blended with at least one pixel value of an underlying object in dependence on the transparency factor. This can allow images in the background to be visible at the same time as underlying objects. For example, the background may have a logo or watermark which is visible, but through which underlying objects may be seen. The transparency factor may be adjustable, for example by the user or under control of the client of the window. In a further embodiment, different transparency factors are assigned to different parts of the background. For example, some parts of the background may be completely transparent, while other parts may have background pixel values which are blended with pixel values of underlying objects.

Preferably, at least one of the objects underlying the background comprises an element of a web page; the element could for example be text, a hyperlink, or other HTML element. In this way, more flexible shapes can be allowed for a control panel, for example to allow a navigation bar to be displayed more efficiently, giving a bigger view of a web page.

30

The method may further comprise displaying a further window which has a background through which underlying objects are visible; hence more than one "transparent window" may be provided. To achieve stacked transparent windows, at least a part of the first window may underlie the further window.

- 4 -

A problem which has been identified pursuant to the present invention is that drawing in a window which lies underneath a window with a transparent background may result in corruption of images in the transparent window. For example, foreground objects in a transparent window may be overwritten by the drawing. The method may therefore
5 further comprise monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

This feature may also be provided independently, and thus the invention also provides a method of drawing in a window, preferably a window which may lie at least partially
10 under an overlying window having a background through which underlying objects are visible, the method comprising drawing in the window and monitoring the drawing so that drawing affecting an overlying window can be corrected, the method being preferably carried out by a receiver/decoder.

15 In order to correct drawing in the overlying window, the method may further comprise determining a window which may be affected by the drawing, and sending a signal (for example, an expose event) instructing a client of the window which may be affected to redraw at least part of that window.

20 The signal may be a Java event; it may also be a message, preferably sent via a function call.

In one embodiment a signal may be sent following each drawing operation. However, it has been discovered pursuant to the present invention that sending signals in this way
25 may lead to a large number of redrawing operations, with the result that a large amount of processing is dedicated to redrawing. According to a preferred embodiment of the invention this problem is overcome by suppressing the sending of a signal for at least one drawing operation. The term "suppressing the sending of a signal", and like terms, as used herein preferably connotes the signal either not being sent or not being received. For
30 example, intercepting and removing a signal in transit between two parties, and not sending the signal in the first place both constitute a suppression of the signal.

The feature may also be provided independently, and thus in another aspect of the invention there is provided a method of drawing in a window, preferably a window which

- 5 -

may lie at least partially under a window having a background through which underlying objects are visible, the method comprising providing a first mode in which a signal (for example, an expose event) is sent following each drawing operation instructing a client of a window which may be affected by the drawing to redraw at least part of that window, and providing a second mode in which the sending of a signal is suppressed.

The above aspect of the invention can provide the advantage that relatively simple drawing operations may be carried out in the first mode, in which case the affected windows will be continually updated, whereas more complex multiple drawing operations may be carried out in the second mode, which may reduce the amount of processing associated with redrawing.

The signal may be sent from a window manager. Alternatively, the signal may be sent from a client of a window, preferably sent by the client which carried out the drawing, and preferably sent to a window manager. The method preferably further comprises making information relating to the transparency of the window available to the window manager, and the method may further comprise storing the information as an attribute of the window. By making information relating to the transparency of a window available to the window manager, operations such as redrawing can be performed in a centralized and efficient manner. Moreover, less work can be required in order to add the feature of transparent windows to an application, and the amount of extra coding required in each application to implement transparent windows can be reduced.

The term "attribute of a window" as used herein preferably connotes some form of data associated with a window, the data being understood by the window manager and typically stored in a block of memory maintained by the window manager and containing other window attributes. Preferably, the method further comprises sending the information to the window manager, preferably in a message or via a function call. The sending of a signal may be suppressed in dependence on the information.

The sending of a signal may be suppressed in dependence on the relationship of the windows. For example, it may be known that windows having a certain relationship to an active window which is drawn in do not overlap with that window, in which case some processing may be saved by ignoring such windows. For example, windows having

the following relationship to the active window could be ignored:

- mother, that is, the window from which the active window was opened
- superiors, that is the mother, its mother etc.
- sisters, that is, windows which were opened from the same window as the active
5 window
- daughters, that is, windows which were opened from the active window
- inferiors, that is, daughters, their daughters, etc.

Alternatively, all windows could be ignored, or all transparent windows could be ignored.

10

The method may further comprise sending a signal following a number of drawing operations, the number preferably being determined in advance of at least one drawing operation, and preferably being greater than 1, 3, 5, 10, 30, 50 or 100. This can be more efficient than redrawing after each drawing operation, but can avoid the situation where,
15 if a large number of drawing operations is to be performed at one go, the overlying window is not redrawn for a significant length of time (which can be aesthetically displeasing). The method may further comprise redrawing at least part of the window which may be affected.

20 In another aspect of the invention there is provided a method of drawing in a windowing system in which drawing in a window is monitored and a signal is sent to at least one other window which may be affected by the drawing, the method further comprising selecting a mode in which a signal to at least one other window is suppressed for at least one drawing operation.

25

The above method may further comprise selecting a mode in which a signal is sent to said at least one other window, following said at least one drawing operation. The signal may be a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

30

The present invention also provides a method of displaying a window, comprising providing a window manager which manages the display of the window, and passing information relating to the transparency of the window to the window manager.

- 7 -

Preferably, the window manager forms part of a virtual machine.

The invention further provides a method of displaying a navigator display, preferably for a web browser, wherein the navigator display comprises at least one window having a transparent background.

The transparent background may generally be located outside the foreground. The method may be carried out by a receiver/decoder, the receiver/decoder preferably being for use with a television set.

10

The invention is applicable to any windowing system, for example for use with a computer. However the invention has particular application with receiver/decoders, and thus the method may be carried out by a receiver/decoder. Receiver/decoders typically have reduced processing power in comparison to personal computers, and thus embodiments of the invention which reduce the processing involved are particularly suited for use with receiver/decoders.

15

In another aspect of the invention there is provided a method of displaying windows on a screen, comprising:

20

displaying first and second windows, at least the first window having a background through which underlying objects are visible;

determining whether the first window overlaps the second window;

drawing in the second window; and

redrawing at least part of the first window if the first window overlaps the second

25

window.

The method may further comprise determining a part of the first window which overlaps the second window, and the step of redrawing may comprise redrawing that part of the first window.

30

In another aspect of the invention there is provided a method of displaying a transparent window and a determined content of this transparent window on an area of a screen, comprising:

determining whether the area of the screen is at least partly used to display a

portion of another window and another content of the other window corresponding to the portion;

drawing at least the portion of the other window and the corresponding other content; and

5 drawing at least a part of the determined content which overlaps the portion of the other window.

According to a related aspect of the invention, there is provided a receiver/decoder comprising means (typically in the form of a processor) for displaying a window on a
10 screen, wherein the window has a background through which underlying objects are visible, the receiver/decoder preferably being for use with a television set.

The receiver/decoder may further comprise means (typically in the form of a processor) for drawing a frame of the window. The displaying means is preferably adapted to
15 display a window which forms part of a screen comprising a window display, and the receiver/decoder preferably further comprises means for combining the window display with a video image. The displaying means is preferably adapted to display the window on a television screen.

20 The receiver/decoder may further comprise:

means (typically in the form of a processor) for defining the size of the window;

means (typically in the form of a processor) for drawing foreground objects in the window; and

25 means (typically in the form of a processor) for arranging the background of the window.

The invention extends to a receiver/decoder comprising:

means (typically in the form of a processor) for displaying a window;

means (typically in the form of a processor) for defining the size of the window;

30 means (typically in the form of a processor) for drawing foreground objects in the window; and

means (typically in the form of a processor) for arranging the background of the window such that objects underlying the background are visible.

The means for drawing foreground objects may be adapted to set foreground pixels to desired values. The foreground objects may comprise any or all of a straight line, curved line, box, circle, triangle and typographical character, and preferably are adapted to be drawn in at least two draw operations. The foreground objects may comprise interactive
5 controls.

The arranging means is preferably adapted to leave at least one pixel value unaltered in a region defining the background. It may be adapted to blend at least one pixel value with a pixel value of an underlying image, in a region defining the background. It may
10 be adapted to leave at least one foreground pixel unaltered.

At least one of the objects underlying the background may comprise an element of a web page.

15 The receiver/decoder may further comprise means (typically in the form of a processor) for displaying a further window which has a background through which underlying objects are visible. At least a part of the first window may underlie the further window.

The receiver/decoder may further comprise means (typically in the form of a processor)
20 for monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

The invention extends to a receiver/decoder, comprising:

means for drawing in a window; and
25 means for monitoring the drawing so that drawing affecting an overlying window can be corrected.

The receiver/decoder may further comprise means for determining a window which may be affected by the drawing, and for sending a signal instructing a client of the window
30 which may be affected to redraw at least part of that window. The signal may be a Java event. It may be sent following each drawing operation.

The receiver/decoder may further comprise means for suppressing the signal for at least one drawing operation.

The invention extends to a receiver/decoder comprising means for drawing in a window, in first and second modes, wherein in the first mode a signal is sent following each drawing operation instructing a client of a window which may be affected by the drawing to redraw at least part of that window, and in the second mode the sending of a signal is suppressed.

The signal may be sent from a window manager. The signal may be sent from a client of a window, and is preferably sent by the client which carried out the drawing, and is preferably sent to a window manager.

10

The receiver/decoder may further comprise means for making information relating to the transparency of the window available to the window manager. It may further comprise means for storing the information as an attribute of the window. It may further comprise means for sending the information to the window manager, preferably in a message or via a function call.

15

The sending of a signal may be suppressed in dependence on the information. The sending of a signal may be suppressed in dependence on the relationship of the windows.

The receiver/decoder may further comprise means for sending a signal following a number of drawing operations, the number preferably being determined in advance of at least one drawing operation, and preferably being greater than 1, 3, 5, 10, 30, 50 or 100.

20

The receiver/decoder may further comprise means for redrawing at least part of the window which may be affected.

25

In an apparatus aspect of the invention there is provided apparatus for displaying a window comprising means for defining the size of the window, means for drawing foreground objects in the window, and means for arranging the background of the window such that objects underlying the background are visible.

30

The apparatus may further comprise means for drawing a frame of the window.

The means for drawing foreground objects may comprise means for setting foreground

pixels to desired values, and the means for arranging the background may be adapted to leave at least one pixel value unaltered, or to blend at least one pixel value with a pixel value of an underlying image, in a region defining the background.

- 5 The apparatus may further comprise means for monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

In another apparatus aspect there is provided apparatus for drawing in a window, preferably a window which may lie at least partially under an overlying window having
10 a background through which underlying objects are visible, comprising means for drawing in the window and means for monitoring the drawing so that drawing affecting an overlying window can be corrected.

The apparatus may further comprise means for determining a window which may be
15 affected by the drawing, and means for sending a signal instructing a client of the window which may be affected to redraw at least part of that window.

The apparatus may be adapted to send a signal following each drawing operation, or the apparatus may be adapted to suppress the sending of a signal for at least one drawing
20 operation.

In another apparatus aspect there is provided apparatus for drawing in a window, preferably a window which may lie at least partially under a window having a background through which underlying objects are visible, the apparatus having a first
25 mode in which a signal is sent following each drawing operation instructing a client of a window which may be affected by the drawing to redraw at least part of that window, and a second mode in which the sending of a signal is suppressed.

The apparatus may be adapted to suppress the sending of a signal in dependence on the
30 relationship of the windows. The apparatus may be adapted to send a signal following a number of drawing operations.

The apparatus may further comprise means for redrawing at least part of the window which may be affected.

In another apparatus aspect there is provided apparatus for drawing in a windowing system in which drawing in a window is monitored and a signal is sent to at least one other window which may be affected by the drawing, the apparatus comprising means for selecting a mode in which a signal to at least one other window is suppressed for at least one drawing operation.

The apparatus may further comprise means for selecting a mode in which a signal is sent to said at least one other window, following said at least one drawing operation. The signal may be a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

In a related aspect, the present invention provides apparatus for drawing in a windowing system, comprising:

- means (typically in the form of a processor) for drawing in a window;
- means (typically in the form of a processor) for monitoring the drawing in the window and for sending a signal to at least one other window which may be affected by the drawing; and
- means (typically in the form of a processor) for selecting a mode in which a signal to at least one other window is suppressed for at least one drawing operation.

The apparatus preferably further comprises means for selecting a mode in which a signal is sent to said at least one other window, following said at least one drawing operation. The signal may be a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

The invention further extends to apparatus for displaying a window, comprising:

- means (typically in the form of a processor) for displaying the window;
- a window manager for managing the display of the window; and
- means (typically in the form of a processor) for passing information relating to the transparency of the window to the window manager.

Preferably, the window manager forms part of a virtual machine.

The invention also extends to apparatus for displaying a navigator display, comprising

- 13 -

means (typically in the form of a processor) for displaying the navigator display in such a way that the navigator display comprises at least one window having a transparent background.

- 5 Any of the above apparatus may be a receiver/decoder, and hence the invention further extends to a receiver/decoder comprising apparatus as aforesaid.

The invention extends to a television system, comprising a receiver/decoder as aforesaid and a television set connectable to the receiver/decoder. It also extends to a television
10 system, comprising a receiver/decoder as aforesaid and a television set connected to the receiver/decoder.

According to the present invention, there is provided in a further aspect a receiver/decoder for displaying windows on a screen, comprising:

- 15 means (typically in the form of a processor) for displaying first and second windows, at least the first window having a background through which underlying objects are visible;

means (typically in the form of a processor) for determining whether the first window overlaps the second window;

- 20 means (typically in the form of a processor) for drawing in the second window;
and

means (typically in the form of a processor) for redrawing at least part of the first window if the first window overlaps the second window.

- 25 According to the present invention there is provided in a further aspect a computer program product for use with a receiver/decoder, said computer program product (under the control of the program) comprising code for displaying a window on a screen, wherein the window has a background through which underlying objects are visible, the receiver/decoder preferably being for use with a television set.

30

The computer program product may be tangibly embodied for example as software stored on a storage medium such as a hard disk or volatile or non-volatile memory, or as a signal.

The computer program product may further comprise code for drawing a frame of the window.

5 Preferably, the displaying code is adapted to display a window which forms part of a screen comprising a window display, and the product further comprises code for combining the window display with a video image. The displaying code may be adapted to display the window on a television screen.

The computer program product may further comprise:

- 10 code for defining the size of the window;
 code for drawing foreground objects in the window; and
 code for arranging the background of the window.

The invention extends to a computer program product comprising:

- 15 code for displaying a window;
 code for defining the size of the window;
 code for drawing foreground objects in the window; and
 code for arranging the background of the window such that objects underlying the background are visible.

20 Preferably, the code for drawing foreground objects is adapted to set foreground pixels to desired values. Preferably, the foreground objects comprise any or all of a straight line, curved line, box, circle, triangle and typographical character, and preferably are adapted to be drawn in at least two draw operations. Preferably, the foreground objects comprise
25 interactive controls.

The arranging code may be adapted to leave at least one pixel value unaltered in a region defining the background. The arranging code may be adapted to blend at least one pixel value with a pixel value of an underlying image, in a region defining the background.

30 The arranging code may be adapted to leave at least one foreground pixel unaltered.

At least one of the objects underlying the background may comprise an element of a web page.

Preferably, the computer program product further comprises code for displaying a further window which has a background through which underlying objects are visible.

At least a part of the first window may underlie the further window.

5

The computer program product may further comprise code for monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

- 10 The invention extends to a computer program product, comprising:
code for drawing in a window; and
code for monitoring the drawing so that drawing affecting an overlying window
can be corrected.

- 15 The computer program product may further comprise code for determining a window which may be affected by the drawing, and for sending a signal instructing a client of the window which may be affected to redraw at least part of that window. The signal may be a Java event. The signal may be sent following each drawing operation.

- 20 Preferably, the computer program product further comprises code for suppressing the signal for at least one drawing operation.

- The invention extends to a computer program product comprising code for drawing in a window, in first and second modes, wherein in the first mode a signal is sent following
25 each drawing operation instructing a client of a window which may be affected by the drawing to redraw at least part of that window, and in the second mode the sending of a signal is suppressed.

- The signal may be sent from a window manager. The signal may be sent from a client
30 of a window, and is preferably sent by the client which carried out the drawing, and is preferably sent to a window manager.

Preferably, the computer program product further comprises code for making information relating to the transparency of the window available to the window manager.

Preferably, the computer program product further comprises code for storing the information as an attribute of the window.

Preferably the computer program product further comprises code for sending the
5 information to the window manager, preferably in a message or via a function call.

The sending of a signal may be suppressed in dependence on the information. The sending of a signal may be suppressed in dependence on the relationship of the windows.

10 The computer program product may further comprise code for sending a signal following a number of drawing operations, the number preferably being determined in advance of at least one drawing operation, and preferably being greater than 1, 3, 5, 10, 30, 50 or 100.

15 The computer program product may further comprise code for redrawing at least part of the window which may be affected.

The invention extends to apparatus for drawing in a windowing system, comprising:
code for drawing in a window;

20 code for monitoring the drawing in the window and for sending a signal to at least one other window which may be affected by the drawing; and

code for selecting a mode in which a signal to at least one other window is suppressed for at least one drawing operation.

25 The apparatus may further comprise code for selecting a mode in which a signal is sent to said at least one other window, following said at least one drawing operation. The signal may be a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

30 The invention extends to apparatus for displaying a window, comprising:

code for displaying the window;

a window manager for managing the display of the window; and

code for passing information relating to the transparency of the window to the window manager.

The window manager may form part of a virtual machine.

The invention further extends to apparatus for displaying a navigator display, comprising code for displaying the navigator display in such a way that the navigator display
5 comprises at least one window having a transparent background.

The invention also extends to a computer program product for displaying windows on a screen, comprising:

code for displaying first and second windows, at least the first window having a
10 background through which underlying objects are visible;
code for determining whether the first window overlaps the second window;
code for drawing in the second window; and
code for redrawing at least part of the first window if the first window overlaps
the second window.

15

The invention also provides a computer readable medium having stored thereon a program for carrying out any of the methods described above, and a computer program product comprising a program for carrying out any of the methods described above.

20 The invention further extends to a computer program product comprising a memory and processor, the memory having stored therein an application, and the processor (under the control of the application) being adapted to carry out the method as aforesaid.

The invention also extends to a computer program product comprising a program for
25 carrying out the method as aforesaid.

The invention yet further extends to a computer readable medium having stored thereon a program for carrying out the method as aforesaid.

30 The invention extends to a computer readable medium having stored thereon a computer program product as aforesaid.

The invention also extends to a signal tangibly embodying a computer program product as aforesaid.

The invention also provides a method and apparatus substantially as described with reference to and as illustrated in the accompanying drawings.

Features of one aspect may be applied to other aspects and method features may be applied in any combination to features relating to any of an apparatus, receiver/decoder, computer program product, or computer readable medium, and *vice versa*. For example, there may be provided apparatus for displaying a navigator display which performs a method of passing information relating to the transparency of a window to a window manager, and so on.

10

Preferred features of the present invention will now be described, purely by way of example, with reference to the accompanying drawings, in which:-

15

Figure 1 shows the architecture of a typical digital television system;

Figure 2 is a block diagram of a receiver/decoder;

Figure 3 shows the architecture of a receiver/decoder;

Figure 4 shows structure of a graphics system;

Figure 5 shows the interrelationship of various windows;

Figure 6 illustrates the principle of transparent windows;

20

Figure 7 illustrates the drawing of an object which underlies a window;

Figure 8 illustrates the management of redrawing in a transparent windows system;

Figure 9 illustrates different modes of redrawing two overlapping windows;

Figure 10 illustrates the relationship between applications and the window manager;

25

Figure 11 illustrates a typical sequence of drawing operations during the redrawing of two windows;

Figure 12 illustrates a screen display of a web browser;

Figure 13 illustrates a further screen display of the web browser;

30

Figure 14 illustrates a further screen display of the web browser; and

Figure 15 illustrates a further screen display of the web browser.

Digital television system

An overview of a digital television system 1 is shown in Figure 1. The invention includes a mostly conventional digital television system 2 that uses the known MPEG-2 compression system to transmit compressed digital signals. In more detail, MPEG-2
5 compressor 3 in a broadcast centre receives a digital signal stream (typically a stream of video signals). The compressor 3 is connected to a multiplexer and scrambler 4 by linkage 5.

The multiplexer 4 receives a plurality of further input signals, assembles the transport
10 stream and transmits compressed digital signals to a transmitter 6 of the broadcast centre via linkage 7, which can of course take a wide variety of forms including telecommunications links. The transmitter 6 transmits electromagnetic signals via uplink 8 towards a satellite transponder 9, where they are electronically processed and broadcast via notional downlink 10 to earth receiver 12, conventionally in the form of a dish owned
15 or rented by the end user. Other transport channels for transmission of the data are of course possible, such as terrestrial broadcast, cable transmission, combined satellite/cable links, telephone networks etc.

The signals received by receiver 12 are transmitted to an integrated receiver/decoder 13
20 owned or rented by the end user and connected to the end user's television set 14. The receiver/decoder 13 decodes the compressed MPEG-2 signal into a television signal for the television set 14. Although a separate receiver/decoder is shown in Figure 1, the receiver/decoder may also be part of an integrated digital television. As used herein, the term "receiver/decoder" includes a separate receiver/decoder, such as a set-top box, and
25 a television having a receiver/decoder integrated therewith.

In a multichannel system, the multiplexer 4 handles audio and video information received from a number of parallel sources and interacts with the transmitter 6 to broadcast the information along a corresponding number of channels. In addition to audiovisual
30 information, messages or applications or any other sort of digital data may be introduced in some or all of these channels interlaced with the transmitted digital audio and video information.

A conditional access system 15 is connected to the multiplexer 4 and the receiver/decoder
35 13, and is located partly in the broadcast centre and partly in the receiver/decoder. It

- 20 -

enables the end user to access digital television broadcasts from one or more broadcast suppliers. A smartcard, capable of deciphering messages relating to commercial offers (that is, one or several television programmes sold by the broadcast supplier), can be inserted into the receiver/decoder 13. Using the receiver/decoder 13 and smartcard, the
5 end user may purchase commercial offers in either a subscription mode or a pay-per-view mode.

As mentioned above, programmes transmitted by the system are scrambled at the multiplexer 4, the conditions and encryption keys applied to a given transmission being
10 determined by the access control system 15. Transmission of scrambled data in this way is well known in the field of pay TV systems. Typically, scrambled data is transmitted together with a control word for descrambling of the data, the control word itself being encrypted by a so-called exploitation key and transmitted in encrypted form.

15 The scrambled data and encrypted control word are then received by the receiver/decoder 13 having access to an equivalent to the exploitation key stored on a smart card inserted in the receiver/decoder to decrypt the encrypted control word and thereafter descramble the transmitted data. A paid-up subscriber will receive, for example, in a broadcast monthly EMM (Entitlement Management Message) the exploitation key necessary to
20 decrypt the encrypted control word so as to permit viewing of the transmission.

An interactive system 16, also connected to the multiplexer 4 and the receiver/decoder 13 and again located partly in the broadcast centre and partly in the receiver/decoder, enables the end user to interact with various applications via a modemmed back channel
25 17. The modemmed back channel may also be used for communications used in the conditional access system 15.

Receiver/decoder

Referring to Figure 2, the elements of the receiver/decoder 13 or set-top box will now be
30 described. The elements shown in this figure will be described in terms of functional blocks.

The receiver/decoder 13 comprises a central processor 20 including associated memory elements and adapted to receive input data from a serial interface 21, a parallel interface

- 21 -

22, a modem 23 (connected to the modem back channel 17 of Fig. 1), and switch contacts 24 on the front panel of the decoder.

5 The receiver/decoder is additionally adapted to receive inputs from an infra-red remote control 25 via a control unit 26 and also possesses two smartcard readers 27, 28 adapted to read bank or subscription smartcards 29, 30 respectively. The subscription smartcard reader 28 engages with an inserted subscription card 30 and with a conditional access unit 29 to supply the necessary control word to a demultiplexer/descrambler 30 to enable the encrypted broadcast signal to be descrambled. The decoder also includes a conventional
10 tuner 31 and demodulator 32 to receive and demodulate the satellite transmission before being filtered and demultiplexed by the unit 30.

As used in this description, an application is preferably a piece of computer code for controlling high level functions of preferably the receiver/decoder 13. For example,
15 when the end user positions the focus of remote control 25 on a button object seen on the screen of the television set 14 and presses a validation key, the instruction sequence associated with the button is run.

An interactive application proposes menus and executes commands at the request of the
20 end user and provides data related to the purpose of the application. Applications may be either resident applications, that is, stored in the ROM (or FLASH or other non-volatile memory) of the receiver/decoder 13, or broadcast and downloaded into the RAM or FLASH memory of the receiver/decoder 13.

25 Applications are stored in memory locations in the receiver/decoder 13 and represented as resource files. The resource files comprise graphic object description unit files, variables block unit files, instruction sequence files, application files and data files.

The receiver/decoder contains memory divided into a RAM volume, a FLASH volume
30 and a ROM volume, but this physical organization is distinct from the logical organization. The memory may further be divided into memory volumes associated with the various interfaces. From one point of view, the memory can be regarded as part of the hardware; from another point of view, the memory can be regarded as supporting or containing the whole of the system shown apart from the hardware.

The receiver/decoder contains five software layers, organized so that the software can be implemented in any receiver/decoder and with any operating system. Referring to Figure 3, the various software layers are Application Layer 50, Application Programming Interface (API) layer 52, Virtual Machine Layer 54, Device Layer Interface 56 and
5 System Software/Hardware Layer 58. The various layers are described below.

Application Layer

The Application Layer 50 encompasses applications that are either resident in or downloaded to the receiver/decoder. They may be interactive applications used by
10 customers, written in, for example, Java, HTML, MHEG-5 or other languages, or they may be applications used by the receiver/decoder to run such applications. This layer is based on a set of open Application Programming Interfaces (APIs) provided by the Virtual Machine layer. This system allows applications to be downloaded to flash or RAM memory in the receiver/decoder on-the-fly or on demand. The application code can
15 be transmitted in compressed or uncompressed format using protocols such as Data Storage Media Command and Control (DSMCC), Network File Server (NFS) or other protocols.

Interactive applications are applications that the user interacts with, for example, to
20 obtain products and services, such as electronic program guides, telebanking applications and games. Various security features are provided for these downloaded applications and data, as follows:

- Nothing can be downloaded to the receiver/decoder without first having been authenticated for the intended network, which prevents any unregistered software
25 from being run in the receiver/decoder. This means that any software running in the receiver/decoder is recognized and has been fully tested.
- A Security Manager limits access of applications to various memory zones, thus assuring data integrity.
- The system can interface with any conditional access system that makes use of
30 secure processors (for example, smart cards inserted in the receiver/decoder).

The following resident applications are used to manage interactive applications:

- Boot. The Boot application is the first application launched when the receiver/decoder is powered on. The Boot application starts the different

"Managers" in the Virtual Machine, the first being the Application Manager.

- Application Manager. The Application Manager manages the interactive applications that are run in the receiver/decoder, that is, it starts, stops, suspends, resumes, handles events and deals with communication between applications. It allows multiple applications to run at once, and thus is involved in the allocation of resources among them. This application is completely transparent to the user.
- SetUp. The purpose of the SetUp application is to configure the receiver/decoder, primarily the first time it is used. It performs actions such as scanning for TV channels, setting the date and time, establishing user preferences, and so on. However, the SetUp application can be used at any time by the user to change the receiver/decoder configuration.
- Zapping. The Zapping application is used to change channels using the Program-up, Program-down and numeric keys. When another form of zapping is used, for example, through a banner (pilot) application, the Zapping application is stopped.

API Layer

The API layer 52 provides high-level utilities for interactive application development. It includes several packages that make up this high-level API. The packages provide all the functionality necessary to run interactive applications. The packages are accessible by the applications.

In a preferred embodiment the API is adapted to run applications written in the Java programming language. Furthermore, it can interpret HTML and other formats, such as MHEG-5. Besides these interpreters, it also includes other packages and service modules that are detachable and extensible as requirements dictate.

Virtual Machine Layer

The Virtual Machine layer 54 is composed of language interpreters and various modules and systems. It consists of everything necessary to receive and execute interactive applications in the receiver/decoder, including the following:

- Language Interpreters. Different interpreters can be installed to conform to the type of applications to be read. These include Java, HTML, MHEG-5 and others.
- Service Information (SI) Engine. The SI Engine loads and monitors common Digital Video Broadcasting (DVB) or Program System Information Protocol

- 24 -

(PSIP) tables and puts them into a cache. It allows access to these tables by applications which need the data contained in them.

- Scheduler. This module allows for pre-emptive, multithreaded scheduling with each thread having its own event queue.
- 5 • Memory Manager. This module manages the access to memory. It also automatically compresses data in memory when necessary and performs automatic garbage collection.
- Event Manager. This module allows events to be triggered according to priority. It manages timer and event grabbing and allows applications to send events to
10 each other.
- Dynamic Linker. This module allows the resolution of addresses arising from native Java functions, loads native methods from a Java class downloaded into RAM and resolves calls from downloaded native codes towards ROM.
- Downloader. This module uses automatic data loading from a remote DSMCC
15 carousel or through the NFS protocol, with downloaded files accessed in the same way as resident ones. Memory clear-up, compression and authentication are also provided.
- Class Manager. This module loads classes and resolves any class referencing problems.
- 20 • File System. This module is compact and optimized to manage a hierarchical file system with multiple ROM, flash, RAM and DSMCC volumes. Flash integrity is guaranteed against any incidents.
- Security Manager. This module authenticates applications and controls the access of applications to sensitive memory and other zones of the set-top box.
- 25 • Graphics System. This system is object-orientated and optimized. It includes graphic window and object management as well as a vectorial font engine with multi-language support.

Furthermore, the DAVIC resource notification model is supported so that client resources
30 are efficiently managed.

Device Layer

The Device Layer 56 includes the Device Manager and devices. Devices are software modules which consist of the logical resources necessary for management of external

- 25 -

events and physical interfaces. The Device Layer manages communication channels between drivers and applications and provides enhanced error exception checking. Some examples of managed devices are: card readers, modems, network, PCMCIA (Personal Computer Memory Card International Association), LED display and so on.

5

Programmers do not have to deal with this layer directly, since the API layer controls the devices from above.

System Software/Hardware Layer

- 10 The System Software/Hardware layer 58 is provided by the manufacturer of the receiver/decoder. Because of the modularity of the system and because services supplied by the OS (such as event scheduling and memory management) are part of the Virtual Machine, the higher layers are not tied to a particular real-time operating system (RTOS) or to a particular processor.

15

Graphic System

- The Graphic System 60, which is part of the Virtual Machine 54 will now be described. Referring to Figure 4, the graphic system graphic system comprises widget tool kit 62, which allows graphic objects such as buttons to be drawn, windowing system 64 and
20 graphic library 68. The windowing system 64 manages a multi-window system in which applications may draw and resize windows. The application which draws a window is referred to as the client of that window.

- As an example, an application may produce a window having a menu of choices in one
25 area of the screen, while a video image is shown across the whole screen behind the window. In this example, referring to Figure 2, the menu is produced by CPU 20 while the video comes from video decoder 34. The two are mixed in the graphic processor 26, under control of CPU 20. More complex arrangements are also possible, with multiple windows each having still or moving images within them.

30

The windowing system maintains a window tree, which defines the relationship between windows. An example of a window tree is shown in Figure 5. Referring to Figure 5, the window 72 from which a particular window 70 has been opened is referred to as the mother, windows 74, 76 which were opened from window 70 are referred to as daughters,

and windows 78, 80 which were opened from the same mother window 72 are referred to as sisters.

Transparent Windows

- 5 In a preferred embodiment, the windowing system manages windows known as transparent windows. A transparent window is a window which does not have a background, that is, through which underlying images can be at least partially seen. The principle of transparent windows will now be described with reference to Figures 6.
- 10 Figure 6a shows two windows 100, 102 which are not superimposed. Window 100 comprises object 104 and opaque background 106; window 102 comprises two objects 108, 110 and opaque background 112. Figure 6b shows the situation when the two windows in Figure 6a are superimposed. In this case, the overlying window completely obscures the underlying window in the area which is common to the two windows.
- 15 Figure 6c shows the same situation as in Figure 6a, but for the case where the overlying window is declared a transparent window. In this case, the two objects 108, 110 are still visible, but the background 112 is now transparent such that window 100 may be seen through it. In the example shown in Figure 6c, the background 112 is simply not drawn.
- 20 Use of transparent windows allows independent designs to be created in different windows, while allowing images in underlying windows to be seen through overlying windows. The present system also allows multiple windows to be displayed, which may consist of a combination of transparent and non-transparent windows.
- 25 It should be noted that a transparent window may have a background which is completely transparent, or it may have a background with a certain transparency factor, in which case the background is blended with the underlying objects according to the transparency factor.
- 30 Specific mechanisms are provided to manage transparent windows. These mechanisms handle the situation where drawing takes beneath a transparent window. As will be explained, such drawing may result in corruption of images in the transparent window unless specific action is taken to prevent this.

- 27 -

Figure 7a shows the situation when a line is drawn underneath a window with an opaque background. In this case, when line 114 is drawn, no changes occur in window 102. Figure 7b shows the situation when window 102 is declared a transparent window. In this case, when line 114 is drawn, it also appears in window 102, with the result that it
5 overwrites the objects 108 and 110 in window 102.

One way of avoiding the overwriting of objects in a transparent window is to prevent drawing from taking place underneath the window. However, in a preferred embodiment, the overwriting of objects in a transparent window is corrected, as described below.

10

Firstly, the area affected by the drawing is determined. This is done by monitoring the maximum and minimum coordinates of the drawing while the drawing takes place. The windows system then searches through the window tree to identify any transparent windows which lie over the affected area. A so-called Expose Event is then sent to the
15 clients of all such windows. The Expose Event informs the clients that the affected area should be redrawn. A client of a window is responsible for the content of that window, and thus the client is able to redraw an overwritten window. The client redraws that part of the window which coincides with the affected area.

20 The above sequence of events is illustrated in Figures 8. Referring to Figure 8a, the area enclosed by rectangle 116 is the area affected by the drawing of line 114. The windows system then identifies window 102 as a transparent window which lies over affected area 116, and an Expose Event is sent to that window. Referring to Figure 8b, the part of the window 102 which is affected by the drawing is indicated by rectangle 118. The area
25 within rectangle 118 is then redrawn by the client. As shown in Figure 8c, the parts of window 102 which were corrupted by the drawing of line 114 are thereby redrawn, so that the objects 108 and 110 are no longer overwritten by line 114.

The above mechanisms allow an application to draw images in a window even if it is
30 overlaid by a transparent window, without corrupting the images in the overlying window.

The number of Expose Events which are sent to the overlying window (that is the number of times that the overlying window is redrawn) is proportional to the number of draw

- 28 -

commands which are generated. A draw command is generated each time a single draw operation takes place. To create a complex image, a large number of draw operations may be necessary, each consisting of, for example, the drawing of a straight or curved line, a box, a circle, or some other object. This can result in a large amount of processing
5 being dedicated to redrawing windows, which may reduce the performance of the system. The situation is made worse if multiple windows are open, particularly multiple overlying transparent windows, each of which may require an Expose Event.

In a preferred embodiment of the present invention, the number of Expose Events which
10 are generated, and hence the number of times that an overlying window is redrawn, is reduced by suppressing the generation of Expose Events in certain circumstances. A parameter is provided, referred to herein as win_over_mode, which specifies the situations in which the generation of Expose Events will be suppressed. The parameter win_over_mode allows client applications to define which windows will be taken into
15 account when deciding to which windows an Expose Event will be sent.

In the preferred embodiment, the win_over_mode parameter may define the following situations with regard to Expose Events:

- 20 - Clip All: normal function in which all windows are taken into account
- Ignore Inferiors: daughters and their daughters are ignored
- Ignore Sisters: sisters and their daughters are ignored
- Ignore Others: all windows which are neither sisters nor daughters are ignored
- Ignore No Background: all transparent windows are ignored
- 25 - Ignore All: all windows are ignored.

Windows which are excluded from the clipping calculation do not receive Expose Events, and thus are not redrawn. For example, it might be known that sister windows never overlie the window in which the drawing takes place. Alternatively, it might be
30 decided to postpone redrawing all transparent windows until the drawing is complete.

Once the drawing is complete, any affected windows may be redrawn. This is done by monitoring the maximum and minimum coordinates of the drawing. When the drawing is complete, the win_over_mode parameter is then set to clip all. A virtual rectangle is

then drawn around all affected areas, using the maximum and minimum coordinates of the drawing. By drawing a virtual window, an Expose Event is generated which is sent to all affected windows in the window tree. The windows which have areas coinciding with the affected area are then redrawn, in the way described above.

5

In summary, greater efficiency can be obtained by deferring the redraw of transparent windows affected by drawing operations performed by an application.

One problem remains, however, as a result of the fact that the windowing system itself
10 may decide to redraw part of a window, for example as a result of the user moving or
resizing the window, or moving or resizing an overlapping window. In this case, without
further information, the windows system always sends Expose Events to the appropriate
windows to ensure a correct result; consequently, more Expose Events are sent, and
therefore more time is spent redrawing windows, than if the client was in control of the
15 process.

With reference to Figures 9, the problem described above and its solution are illustrated.
Figure 9a shows a typical configuration of two windows: a transparent Window A 150
and an opaque Window B 152 which it partially overlaps.

20

In Figure 9b, which corresponds to a variant of the preferred embodiment, a typical
sequence of Expose Events is illustrated, following a decision by the window manager
to redraw part of Window B. Expose Events 160 and 162 are sent to Window B 152 and
Window A 150 respectively. Window B, receiving the Expose Event 160, is redrawn,
25 thereby generating a further Expose Event 164, which is sent to Window A. Window A,
receiving two Expose Events, is therefore redrawn twice.

In Figure 9c, illustrating the preferred embodiment, information regarding the
optimisation of Expose Events is made available to the window manager by means of the
30 window attributes of Window A 150. Taking into account this information, a further
Expose Event 170 is initially only sent to Window B 152. The subsequent redrawing of
Window B then produces the required Expose Event 172, sent to Window A 150 so that
it is redrawn only once.

The window attribute which defines the optimisation of the Expose Events in the preferred embodiment has the name of `no_bg_win_management` (for no background windows management), and takes the values of:

5 - Automatic: each draw operation in the window causes events to be sent to transparent windows.

 - LetMeDoIt: once the various redraws have finished, the client calls a function (SendEventToNoBgWindow) to specify the extent of the redraws (typically
10 supplying four coordinates specifying the corners of a bounding rectangle); this function determines the zones of the transparent windows to which it must then send Expose Events, thus provoking their redraw in turn. Information regarding the optimisation of Expose Events may of course be made available to the window manager via other means, such as function calls, and so on.

15

 In can be seen from the above that many advantages, such as overall efficiency, ease of use, effective code reuse (due to the transparent windows management code being present once in the window manager instead of several times in different applications), centralized control, speed, and so on are afforded by integrating the management of
20 transparent windows into the window manager.

 With reference to Figure 10, the relationship between the window manager and the clients (applications) of the windows is now described in more detail.

25 Each window 2010, 2012, 2014 on the screen 2000 has a corresponding window definition data block 2020, 2022, 2024 respectively. Each window is also associated with a particular client (application); in this case, the window 2010 and its corresponding window definition data block 2020 are associated with a first application 2030, and windows 2012, 2014 and their corresponding window definition data blocks 2022, 2024
30 are associated with the application 2032. A window manager 2050 manages the whole system, and has access to all of the window definition data blocks 2020, 2022, 2024 and can communicate with each application 2030, 2032 by the use of events, messages and/or function calls.

- 31 -

The window definition data blocks contain the various attributes regarding the state of the corresponding window, such as the size, position, border style and modes of behaviour, and including the no_bg_win_management attribute described above, for example. Information regarding the transparency of a window can also be stored in the window definition data block.

In the preferred embodiment, information regarding the transparency of a window 2010 is stored in the window definition data block 2020 by means of a function call 2042 to the window manager 2050. In variants of the preferred embodiment, however, the information is sent to the window manager via other means, or stored directly in the window definition data block 2020, indicated schematically by the link 2040. In addition to being able to communicate with the window manager 2050, applications can communicate directly between each other, indicated schematically by the link 2044.

With reference to Figures 11, an example in which two overlapping windows are redrawn will now be described, illustrating the above features of the deferred redrawing of overlying transparent windows and the integration of the management of transparent windows into the window manager. In this example, like parts are denoted by like numerals.

The contents of the two windows C 2100 and D 2102 are shown in Figure 11a. Window C 2100 is transparent, and comprises a triangle 2110 and a rectangle 2112. Window D 2102 is opaque, and comprises three vertical lines 2114, 2116, 2118.

When displayed on screen, with reference to Figure 11b, the two windows overlap as shown, with Window C 2100 overlying Window D 2102, partially obscuring the contents of Window D.

With reference to Figure 11c, a refresh event occurs in relation to Window D, causing the opaque background 2120 and border 2122 of Window D 2102 to be redrawn, overwriting the overlying transparent Window C 2100. In addition, in the first drawing operation performed by the client (application), the first of the three vertical lines 2124 is drawn. It is noted that, in the present example, the window manager does not redraw the overlying window after drawing the 'system' areas of the window (the border and

background, for example), but in variants of the preferred embodiment, such a redraw does take place before the client (application) is requested to redraw the 'user' areas.

5 The simplest case, where overlying windows are redrawn after each subsequent drawing operation, is illustrated in Figures 11d and 11e. In Figure 11d, the overlying Window C 2100 is redrawn, following the sending of an Expose event to that window. With reference to Figure 11e, the next drawing operation then takes place, with the drawing of the vertical line 2126. As discussed above, in this case, the overlying window will eventually be redrawn three times in total, two more than are necessary.

10

In the case where the redrawing of the overlying window is deferred, as described above and with reference to Figure 11f, all three vertical lines 2124, 2126, 2128 are drawn in Window D 2102 before the overlying Window C 2100 is redrawn, after which the windows have their correct appearance as illustrated in Figure 11b.

15

A further example in which an object to be drawn requires ten draw commands will now be described, again highlighting the feature of deferred redrawing of overlying transparent windows. As before, the object is to be drawn in a environment containing transparent windows.

20

In accordance with the methods described above, one of two procedures may be chosen. The first procedure is as follows:

1. Set the win_over_mode parameter to Ignore No Background
- 25 2. Perform the ten drawing operations. No Expose Events are generated, since all transparent windows are ignored.
3. Set the win_over_mode parameter to Clip All
4. Draw a virtual rectangle which encompasses all areas which have been drawn in. This operation generates an Expose Event which is sent to the affected windows.

30

The second, shorter, procedure which may be chosen is as follows:

1. Set the no_bg_win_management attribute to LetMeDoIt
2. Perform the ten drawing operations. No Expose Events are generated, since all

transparent windows are ignored.

3. Call `SentEventToNoBgWindow()`, passing as a parameter the physical extent of the ten drawing operations. Expose Events are then sent to the affected windows.

- 5 It is noted that both procedures achieve the same result.

In the above example, rather than the affected windows being redrawn ten times, as would be the case if an Expose Event were generated for each draw command, the windows are only redrawn once. In this way, considerable savings in processing may be made.

In a variant of the preferred embodiment, a further method of redrawing windows is employed. This method comprises sending a signal to redraw overlying windows following a specific number of drawing operations, where the number is usually determined in advance of the drawing operations. With reference to the above example where a total of ten drawing operations were required, if the number chosen was 5, any overlying windows would be redrawn after the 5th and 10th drawing operations.

This method of redrawing offers the advantage of redrawing less frequently than if the redraw occurred after each drawing operation, but offers a further advantage of keeping the screen refreshed relatively frequently in the event of a large number of consecutive drawing operations being performed. The number of drawing operations to wait for before refreshing can be almost any number, from 1 upwards (but typically would be between 1 and 1000), depending on the current processor load, for example. The handling of the counting and redrawing may be handled either by the client or by the window manager.

Use of the optimizations described above with regard to the deferral of redrawing of transparent windows allows an application to use transparent windows with a considerable reduction in the amount of processing involved. This is of particular advantage in environments such as receiver/decoders where processing power may be limited.

Further advantages particular to receiver/decoders are also afforded. For example, the

- 34 -

use of transparent windows can typically allow a smaller windowing display to be provided, since the windows can be stacked on top of each other more effectively. With the windowing display typically in use being overlaid on a video image (by the graphics processor described above, which takes inputs both from the video decoder and from the windowing system), a smaller windowing display is highly preferential, since less of the video image would then be obscured.

Furthermore, the method of displaying transparent windows typically employed (in contrast to blending methods) can result in a higher contrast images when displayed on a television (which cannot generally support as high a resolution as a computer monitor, for example).

With reference to Figures 12 to 15, a further example will be described, concerning a particular application of the transparent windowing system described above: namely, an Internet browser application for a receiver/decoder.

Figure 12 shows a screenshot of the main screen navigator display of the Internet browser. The main screen shows a vertical chain 1100 which includes the main menu which includes various buttons as a linked list. The buttons are linked by link elements of the chain. The buttons shown in the chain 1100 of Figure 12 include the RELOAD/STOP button 1110, the PREVIOUS button 1120, NEXT button 1130, HISTORY button 1140, BOOKMARK button 1150, SETTINGS button 1160 and the QUIT button 1170.

The main menu chain 1100 is arranged to be superimposed on the hyper-text markup language (HTML) document to be displayed on the screen 1101. In Figure 12, no HTML document is displayed and the screen 1101 is blank apart from the main menu chain 1100.

The user has a controller with which he can navigate between objects on the screen 1101, highlight objects and select objects. In the present example, the controller used is a television remote control. When an HTML page is displayed, any button of the remote control calls the chain 1100 (toolbar). The chain 1100 can also be turned on and off by

the user.

Figure 13 shows the screen of Figure 12 having an HTML document open. In Figure 13, the PREVIOUS button 1120 is now highlighted.

5

Figure 14 shows how, when the highlight is on the PREVIOUS button 1120 a "tooltip" comprising a text box 1122 appears on the screen. In the present example, the tooltip appears as soon as the relevant icon is highlighted. The preferences could be set so that the tooltip appears after a delay when the button is highlighted. The text box 1122 includes the word "previous" to indicate the function of the PREVIOUS button 1120. By activating the PREVIOUS button, by pressing the SELECT key 1186, the browser moves to the previous page viewed.

A "widget set", used to draw the above navigation chain (amongst other things) will now be described, before returning to the discussion of Figures 12 to 14. One particular application of such a widget set is to provide widgets in a GUI display of a receiver/decoder for digital television. Amongst other things, the widget set contains a set of primitive widget classes 410 including, amongst others, window and dialogue box frames, a slider control, a push button, a check box, a text field, and a text editing box.

20

Each of the widget classes implements methods and event handlers to control operation of the widget. The widget classes may also contain methods for drawing some parts of the widget. However, in order to provide a particular appearance or "look" for the widget, the widget classes invoke drawing methods of a look object class with which the widget class is associated.

25

A look includes, amongst other things, a definition of colours, so that any widget instance which is associated with a particular look instance will use the colours defined in that look instance. In one embodiment, a look defines several colours, such as black, dark gray, middle gray, light gray, very light gray, white, and highlight colour. Significantly, and in addition, it also defines a transparent colour.

30

Figure 15 shows in more detail the top-level chain of navigator function buttons (1410)

described above, with a portion of a web page visible beneath (1411).

As mentioned previously, the main screen displays a vertical bar (main menu) (1410) which is composed of a series of buttons. A button is effectively a rectangular or square zone (for example, 32x32 pixels). When the graphics cursor enters a zone, that zone (window) receives the focus.

In the light of the above, one can summarize that each zone is a window, and, as is observed from Figure 15, the background of each of these windows is arranged to be transparent, consequently allowing each button of the toolbar (comprising an interactive foreground object) to have an apparently non-rectangular shape when overlaid on the Internet browser. Apart from allowing the designer of the Internet browser more flexibility, this can allow a more compact arrangement of the buttons with consequent advantages.

The structure and function of the above Internet browser and navigation toolbar are discussed in more detail in European Patent Application No. 00300832.3 in the name of Canal+ Technologies Societe Anonyme.

It will be understood that the present invention has been described above purely by way of example, and modifications of detail can be made within the scope of the invention.

Each feature disclosed in the description, and (where appropriate) the claims and drawings may be provided independently or in any appropriate combination.

Reference numerals appearing in the claims are by way of illustration only and shall have no limiting effect on the scope of the claims.

CLAIMS

1. A method of displaying a window on a screen wherein the window has a background through which underlying objects are visible, the method being carried out
5 by a receiver/decoder, the receiver/decoder preferably being for use with a television set.
2. A method according to Claim 1, further comprising drawing a frame of the window.
- 10 3. A method according to Claim 1 or 2, wherein the window forms part of a screen comprising a window display and the method further comprises combining the window display with a video image.
4. A method according to any of the preceding claims, further comprising
15 displaying the window on a television screen.
5. A method according to any of the preceding claims, further comprising the steps of:
defining the size of the window;
20 drawing foreground objects in the window; and
arranging the background of the window.
6. A method of displaying a window, carried out by a receiver/decoder and comprising the steps of :
25 defining the size of the window;
drawing foreground objects in the window; and
arranging the background of the window such that objects underlying the background are visible.
- 30 7. A method according to Claim 5 or 6, wherein the step of drawing foreground objects comprises setting foreground pixels to desired values.
8. A method according to any of Claims 5 to 7, wherein the foreground objects comprise any or all of a straight line, curved line, box, circle, triangle and

typographical character, and preferably are drawn in at least two draw operations.

9. A method according to any of Claims 5 to 8, wherein the foreground objects comprise interactive controls.

5

10. A method according to any of Claims 5 to 9, wherein the step of arranging the background comprises leaving at least one pixel value unaltered in a region defining the background.

10 11. A method according to any of Claims 5 to 10, wherein the step of arranging the background comprises blending at least one pixel value with a pixel value of an underlying image, in a region defining the background.

12. A method according to any of Claims 5 to 11, wherein the step of
15 arranging the background comprises leaving at least one foreground pixel unaltered.

13. A method according to any of Claims 5 to 12, wherein at least one of the objects underlying the background comprises an element of a web page.

20 14. A method according to any of Claims 5 to 13, further comprising displaying a further window which has a background through which underlying objects are visible.

25 15. A method according to Claim 14, wherein at least a part of the first window underlies the further window.

16. A method according to any of the preceding claims, further comprising monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

30

17. A method of drawing in a window, the method comprising drawing in the window and monitoring the drawing so that drawing affecting an overlying window can be corrected, the method being carried out by a receiver/decoder.

18. A method according to Claim 16 or 17, further comprising determining a window which may be affected by the drawing, and sending a signal instructing a client of the window which may be affected to redraw at least part of that window.

5 19. A method according to Claim 18, wherein the signal is a Java event.

20. A method according to Claim 18 or 19, wherein the signal is sent following each drawing operation.

10 21. A method according to Claim 20, further comprising suppressing the signal for at least one drawing operation.

22. A method of drawing in a window, the method comprising providing a first mode in which a signal is sent following each drawing operation instructing a client
15 of a window which may be affected by the drawing to redraw at least part of that window, and providing a second mode in which the sending of a signal is suppressed.

23. A method according to any of Claims 18 to 22, wherein the signal is sent from a window manager.

20 24. A method according to any of Claims 18 to 22, wherein the signal is sent from a client of a window, preferably sent by the client which carried out the drawing, and preferably sent to a window manager.

25 25. A method according to Claim 23 or 24, further comprising making information relating to the transparency of the window available to the window manager.

26. A method according to Claim 25, further comprising storing the information as an attribute of the window.

30 27. A method according to Claim 25 or 26, further comprising sending the information to the window manager, preferably in a message or via a function call.

28. A method according to any of Claims 25 to 27, wherein the sending of a

signal is suppressed in dependence on the information.

29. A method according to any of Claims 18 to 26, wherein the sending of a signal is suppressed in dependence on the relationship of the windows.

5

30. A method according to any of Claims 18 to 29 further comprising sending a signal following a number of drawing operations, the number preferably being determined in advance of at least one drawing operation, and preferably being greater than 1, 3, 5, 10, 30, 50 or 100.

10

31. A method according to any of Claims 16 to 30, further comprising redrawing at least part of the window which may be affected.

32. A method of drawing in a windowing system in which drawing in a window is monitored and a signal is sent to at least one other window which may be affected by the drawing, the method further comprising selecting a mode in which a signal to at least one other window is suppressed for at least one drawing operation.

15

33. A method according to Claim 32, further comprising selecting a mode in which a signal is sent to said at least one other window, following said at least one drawing operation.

20

34. A method according to Claim 32 or 33 wherein the signal is a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

25

35. A method of displaying a window, comprising providing a window manager which manages the display of the window, and passing information relating to the transparency of the window to the window manager.

30

36. A method according to Claim 35, wherein the window manager forms part of a virtual machine.

37. A method of displaying a navigator display, preferably for a web browser,

wherein the navigator display comprises at least one window having a transparent background.

38. A method according to Claim 37, wherein the method is carried out by a receiver/decoder, the receiver/decoder preferably being for use with a television set.

39. A method of displaying windows on a screen, the method being carried out by a receiver/decoder and comprising the steps of: –
displaying first and second windows, at least the first window having a background through which underlying objects are visible;
determining whether the first window overlaps the second window;
drawing in the second window; and
redrawing at least part of the first window if the first window overlaps the second window.

40. A receiver/decoder comprising means (typically in the form of a processor) for displaying a window on a screen, wherein the window has a background through which underlying objects are visible, the receiver/decoder preferably being for use with a television set.

41. A receiver/decoder according to Claim 41, further comprising means (typically in the form of a processor) for drawing a frame of the window.

42. A receiver/decoder according to Claim 40 or 41, wherein the displaying means is adapted to display a window which forms part of a screen comprising a window display, and further comprising means for combining the window display with a video image.

43. A receiver/decoder according to any of Claims 40 to 42, wherein the displaying means is adapted to display the window on a television screen.

44. A receiver/decoder according to any of Claims 40 to 43, further comprising:
means (typically in the form of a processor) for defining the size of the window;

means (typically in the form of a processor) for drawing foreground objects in the window; and

means (typically in the form of a processor) for arranging the background of the window.

5

45. A receiver/decoder comprising:

means (typically in the form of a processor) for displaying a window;

means (typically in the form of a processor) for defining the size of the window;

means (typically in the form of a processor) for drawing foreground objects in the

10 window; and

means (typically in the form of a processor) for arranging the background of the window such that objects underlying the background are visible.

46. A receiver/decoder according to Claim 44 or 45, wherein the means for
15 drawing foreground objects is adapted to set foreground pixels to desired values.

47. A receiver/decoder according to any of Claims 44 to 46, wherein the foreground objects comprise any or all of a straight line, curved line, box, circle, triangle and typographical character, and preferably are adapted to be drawn in at least two draw
20 operations.

48. A receiver/decoder according to any of Claims 44 to 47, wherein the foreground objects comprise interactive controls.

25 49. A receiver/decoder according to any of Claims 44 to 48, wherein the arranging means is adapted to leave at least one pixel value unaltered in a region defining the background.

50. A receiver/decoder according to any of Claims 44 to 49, wherein the
30 arranging means is adapted to blend at least one pixel value with a pixel value of an underlying image, in a region defining the background.

51. A receiver/decoder according to any of Claims 44 to 50, wherein the arranging means is adapted to leave at least one foreground pixel unaltered.

52. A receiver/decoder according to any of Claims 44 to 51, wherein at least one of the objects underlying the background comprises an element of a web page.

53. A receiver/decoder according to any of Claims 44 to 52, further
5 comprising means (typically in the form of a processor) for displaying a further window which has a background through which underlying objects are visible.

54. A receiver/decoder according to Claim 53, wherein at least a part of the first window underlies the further window.

10

55. A receiver/decoder according to any of the preceding claims, further comprising means (typically in the form of a processor) for monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

15

56. A receiver/decoder, comprising:
means for drawing in a window; and
means for monitoring the drawing so that drawing affecting an overlying window can be corrected.

20

57. A receiver/decoder according to Claim 55 or 56, further comprising means for determining a window which may be affected by the drawing, and for sending a signal instructing a client of the window which may be affected to redraw at least part of that window.

25

58. A receiver/decoder according to Claim 57, wherein the signal is a Java event.

59. A receiver/decoder according to Claim 57 or 58, wherein the signal is sent
30 following each drawing operation.

60. A receiver/decoder according to Claim 59, further comprising means for suppressing the signal for at least one drawing operation.

61. A receiver/decoder comprising means for drawing in a window, in first and second modes, wherein in the first mode a signal is sent following each drawing operation instructing a client of a window which may be affected by the drawing to redraw at least part of that window, and in the second mode the sending of a signal is suppressed.

62. A receiver/decoder according to any of Claims 57 to 61, wherein the signal is sent from a window manager.

63. A receiver/decoder according to any of Claims 57 to 61, wherein the signal is sent from a client of a window, and is preferably sent by the client which carried out the drawing, and is preferably sent to a window manager.

64. A receiver/decoder according to Claim 62 or 63, further comprising means for making information relating to the transparency of the window available to the window manager.

65. A receiver/decoder according to Claim 64, further comprising means for storing the information as an attribute of the window.

66. A receiver/decoder according to Claim 64 or 65, further comprising means for sending the information to the window manager, preferably in a message or via a function call.

67. A receiver/decoder according to any of Claims 64 to 66, wherein the sending of a signal is suppressed in dependence on the information.

68. A receiver/decoder according to any of Claims 57 to 66, wherein the sending of a signal is suppressed in dependence on the relationship of the windows.

69. A receiver/decoder according to any of Claims 57 to 68, further comprising means for sending a signal following a number of drawing operations, the number preferably being determined in advance of at least one drawing operation, and preferably being greater than 1, 3, 5, 10, 30, 50 or 100.

70. A receiver/decoder according to any of Claims 55 to 69, further comprising means for redrawing at least part of the window which may be affected.

71. Apparatus for drawing in a windowing system, comprising:
5 means (typically in the form of a processor) for drawing in a window;
means (typically in the form of a processor) for monitoring the drawing in the window and for sending a signal to at least one other window which may be affected by the drawing; and
means (typically in the form of a processor) for selecting a mode in which a signal
10 to at least one other window is suppressed for at least one drawing operation.

72. Apparatus according to Claim 71, further comprising means for selecting a mode in which a signal is sent to said at least one other window, following said at least one drawing operation.

15

73. Apparatus according to Claim 71 or 72 wherein the signal is a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

20

74. Apparatus for displaying a window, comprising:
means (typically in the form of a processor) for displaying the window;
a window manager for managing the display of the window; and
means (typically in the form of a processor) for passing information relating to the transparency of the window to the window manager.

25

75. Apparatus according to Claim 74, wherein the window manager forms part of a virtual machine.

76. Apparatus for displaying a navigator display, comprising means (typically
30 in the form of a processor) for displaying the navigator display in such a way that the navigator display comprises at least one window having a transparent background.

77. A receiver/decoder comprising apparatus according to Claim 76.

78. A television system, comprising a receiver/decoder according to Claim 77 and a television set connectable to the receiver/decoder.

79. A television system, comprising a receiver/decoder according to Claim 77
5 and a television set connected to the receiver/decoder.

80. A receiver/decoder for displaying windows on a screen, comprising:
means (typically in the form of a processor) for displaying first and second
windows, at least the first window having a background through which underlying
10 objects are visible;
means (typically in the form of a processor) for determining whether the first
window overlaps the second window;
means (typically in the form of a processor) for drawing in the second window;
and
15 means (typically in the form of a processor) for redrawing at least part of the first
window if the first window overlaps the second window.

81. A computer program product for use with a receiver/decoder, said
computer program product (under the control of the program) comprising code for
20 displaying a window on a screen, wherein the window has a background through which
underlying objects are visible, the receiver/decoder preferably being for use with a
television set.

82. A computer program product according to Claim 81, further comprising
25 code for drawing a frame of the window.

83. A computer program product according to Claim 81 or 82, wherein the
displaying code is adapted to display a window which forms part of a screen comprising
a window display, and further comprising code for combining the window display with
30 a video image.

84. A computer program product according to any of Claims 81 to 83, wherein
the displaying code is adapted to display the window on a television screen.

85. A computer program product according to any of Claims 81 to 84, further comprising:

- code for defining the size of the window;
- code for drawing foreground objects in the window; and
- 5 code for arranging the background of the window.

86. A computer program product comprising:

- code for displaying a window;
- code for defining the size of the window;
- 10 code for drawing foreground objects in the window; and
- code for arranging the background of the window such that objects underlying the background are visible.

87. A computer program product according to Claim 85 or 86, wherein the
15 code for drawing foreground objects is adapted to set foreground pixels to desired values.

88. A computer program product according to any of Claims 84 to 87, wherein the foreground objects comprise any or all of a straight line, curved line, box, circle, triangle and typographical character, and preferably are adapted to be drawn in at least
20 two draw operations.

89. A computer program product according to any of Claims 84 to 88, wherein the foreground objects comprise interactive controls.

25 90. A computer program product according to any of Claims 84 to 89, wherein the arranging code is adapted to leave at least one pixel value unaltered in a region defining the background.

91. A computer program product according to any of Claims 84 to 90, wherein
30 the arranging code is adapted to blend at least one pixel value with a pixel value of an underlying image, in a region defining the background.

92. A computer program product according to any of Claims 84 to 91, wherein the arranging code is adapted to leave at least one foreground pixel unaltered.

93. A computer program product according to any of Claims 84 to 92, wherein at least one of the objects underlying the background comprises an element of a web page.

5 94. A computer program product according to any of Claims 84 to 93, further comprising code for displaying a further window which has a background through which underlying objects are visible.

10 95. A computer program product according to Claim 94, wherein at least a part of the first window underlies the further window.

15 96. A computer program product according to any of the preceding claims, further comprising code for monitoring drawing in a further window so that drawing in the further window affecting said window can be corrected.

97. A computer program product, comprising:
code for drawing in a window; and
code for monitoring the drawing so that drawing affecting an overlying window can be corrected.

20 98. A computer program product according to Claim 96 or 97, further comprising code for determining a window which may be affected by the drawing, and for sending a signal instructing a client of the window which may be affected to redraw at least part of that window.

25 99. A computer program product according to Claim 98, wherein the signal is a Java event.

30 100. A computer program product according to Claim 98 or 99, wherein the signal is sent following each drawing operation.

101. A computer program product according to Claim 100 further comprising code for suppressing the signal for at least one drawing operation.

102. A computer program product comprising code for drawing in a window, in first and second modes, wherein in the first mode a signal is sent following each drawing operation instructing a client of a window which may be affected by the drawing to redraw at least part of that window, and in the second mode the sending of a signal is
5 suppressed.

103. A computer program product according to any of Claims 98 to 102, wherein the signal is sent from a window manager.

104. A computer program product according to any of Claims 98 to 102, wherein the signal is sent from a client of a window, and is preferably sent by the client which carried out the drawing, and is preferably sent to a window manager.

105. A computer program product according to Claim 103 or 104, further
15 comprising code for making information relating to the transparency of the window available to the window manager.

106. A computer program product according to Claim 105, further comprising code for storing the information as an attribute of the window.

20 107. A computer program product according to Claim 105 or 106, further comprising code for sending the information to the window manager, preferably in a message or via a function call.

25 108. A computer program product according to any of Claims 105 to 107, wherein the sending of a signal is suppressed in dependence on the information.

109. A computer program product according to any of Claims 98 to 107, wherein the sending of a signal is suppressed in dependence on the relationship of the
30 windows.

110. A computer program product according to any of Claims 98 to 109, further comprising code for sending a signal following a number of drawing operations, the number preferably being determined in advance of at least one drawing operation, and

- 50 -

preferably being greater than 1, 3, 5, 10, 30, 50 or 100.

111. A computer program product according to any of Claims 96 to 110, further comprising code for redrawing at least part of the window which may be affected.

5

112. Apparatus for drawing in a windowing system, comprising:

code for drawing in a window;

code for monitoring the drawing in the window and for sending a signal to at least one other window which may be affected by the drawing; and

10 code for selecting a mode in which a signal to at least one other window is suppressed for at least one drawing operation.

113. Apparatus according to Claim 112, further comprising code for selecting a mode in which a signal is sent to said at least one other window, following said at least
15 one drawing operation.

114. Apparatus according to Claim 112 or 113 wherein the signal is a signal instructing a client of a window that may be affected by the drawing to redraw at least part of that window.

20

115. Apparatus for displaying a window, comprising:

code for displaying the window;

a window manager for managing the display of the window; and

25 code for passing information relating to the transparency of the window to the window manager.

116. Apparatus according to Claim 115, wherein the window manager forms part of a virtual machine.

30 117. Apparatus for displaying a navigator display, comprising code for displaying the navigator display in such a way that the navigator display comprises at least one window having a transparent background.

118. A computer program product for displaying windows on a screen,

comprising:

code for displaying first and second windows, at least the first window having a background through which underlying objects are visible;

code for determining whether the first window overlaps the second window;

5 code for drawing in the second window; and

code for redrawing at least part of the first window if the first window overlaps the second window.

119. A computer program product comprising a memory and processor, the
10 memory having stored therein an application, and the processor (under the control of the application) being adapted to carry out the method as claimed in any of Claims 1 to 39.

120. A computer program product comprising a program for carrying out the method as claimed in any of Claims 1 to 39.

15

121. A computer readable medium having stored thereon a program for carrying out the method as claimed in any of Claims 1 to 39.

122. A computer readable medium having stored thereon a computer program
20 product as claimed in any of Claims 80 to 120.

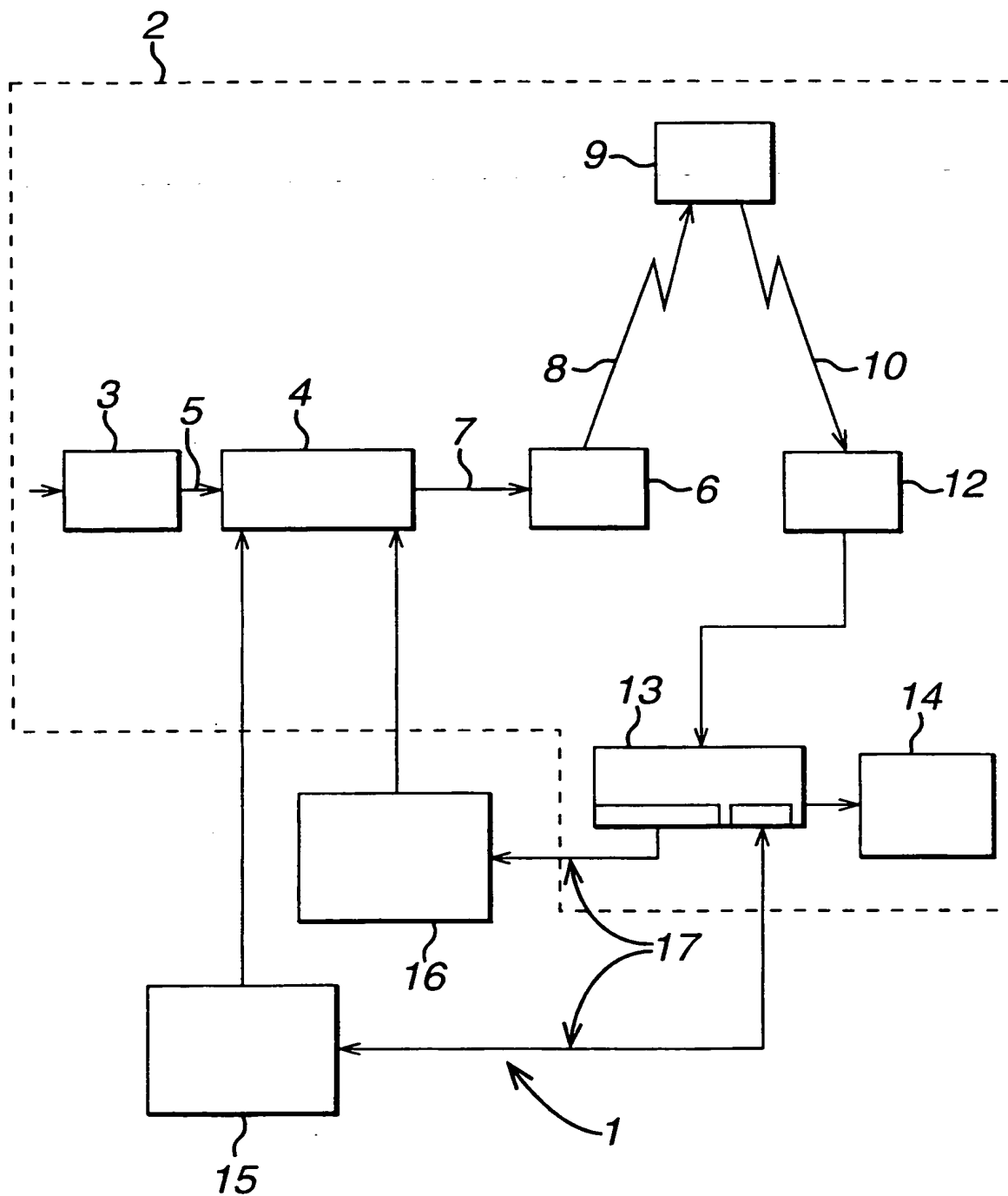
123. A signal tangibly embodying a computer program product as claimed in any of Claims 80 to 120.

25 124. A method substantially as described with reference to and illustrated in the accompanying drawings.

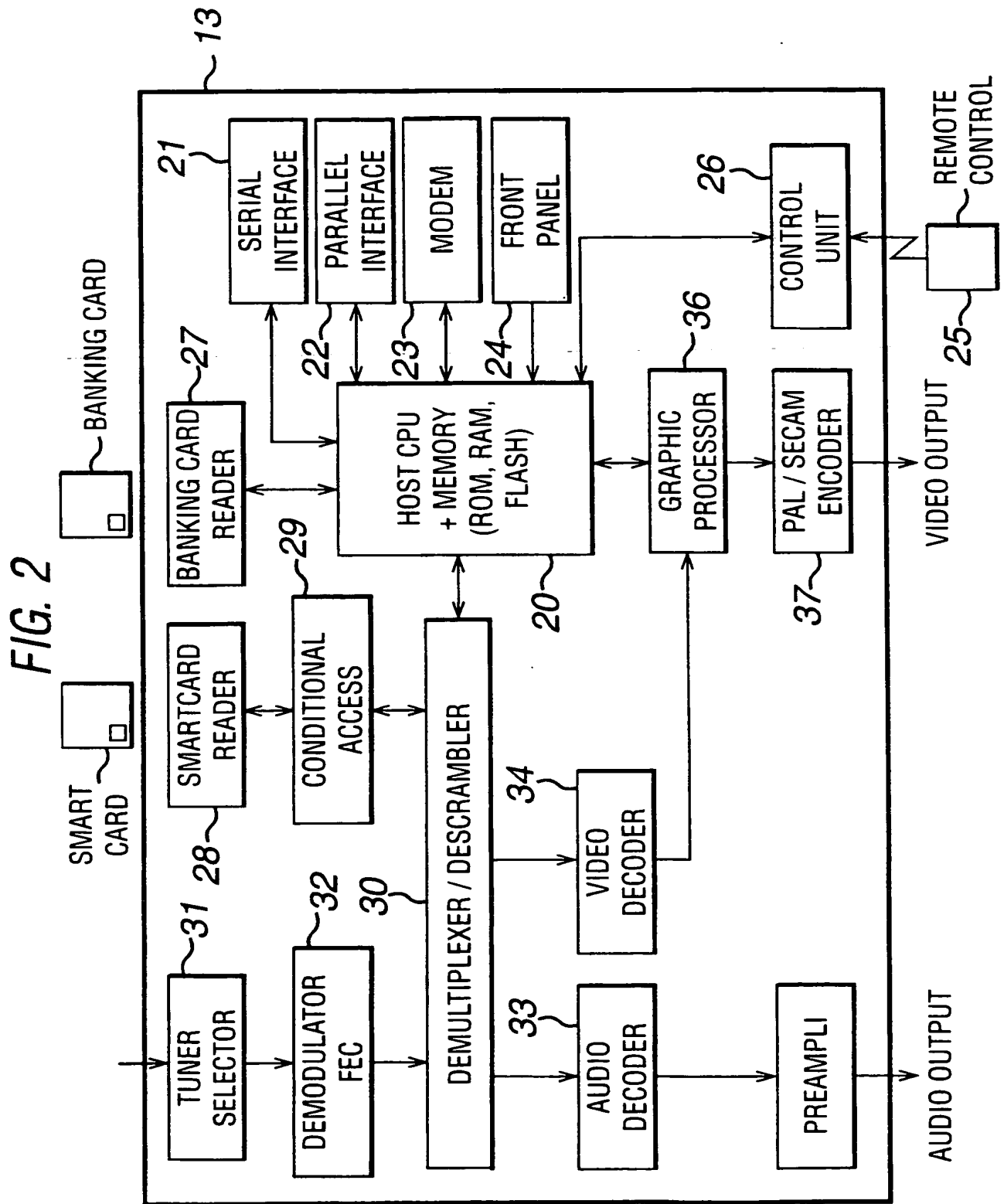
125. Apparatus substantially as herein described with reference to and as illustrated in the accompanying drawings.

1/16

FIG. 1

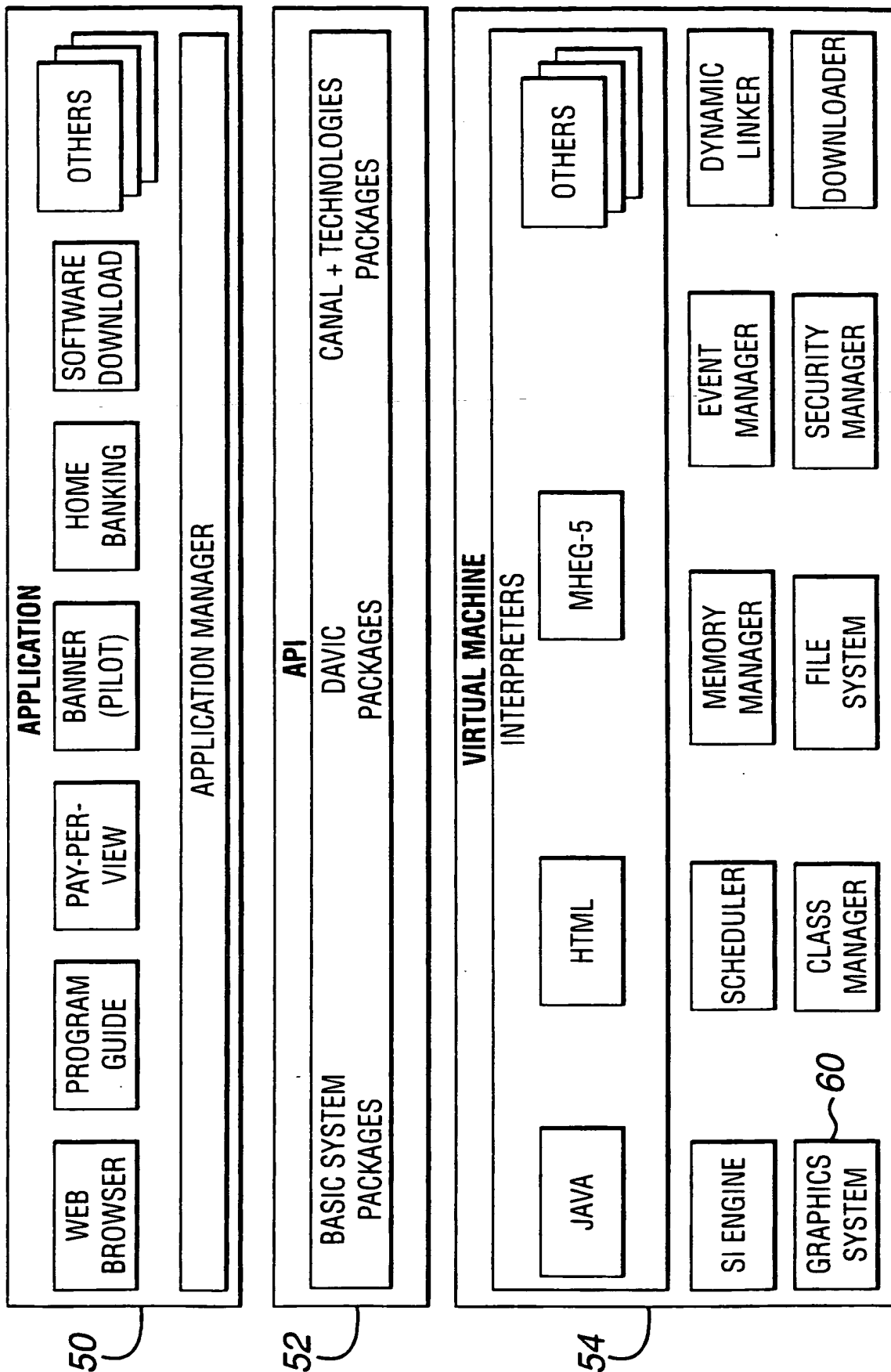


2/16



3/16

FIG. 3



4/16

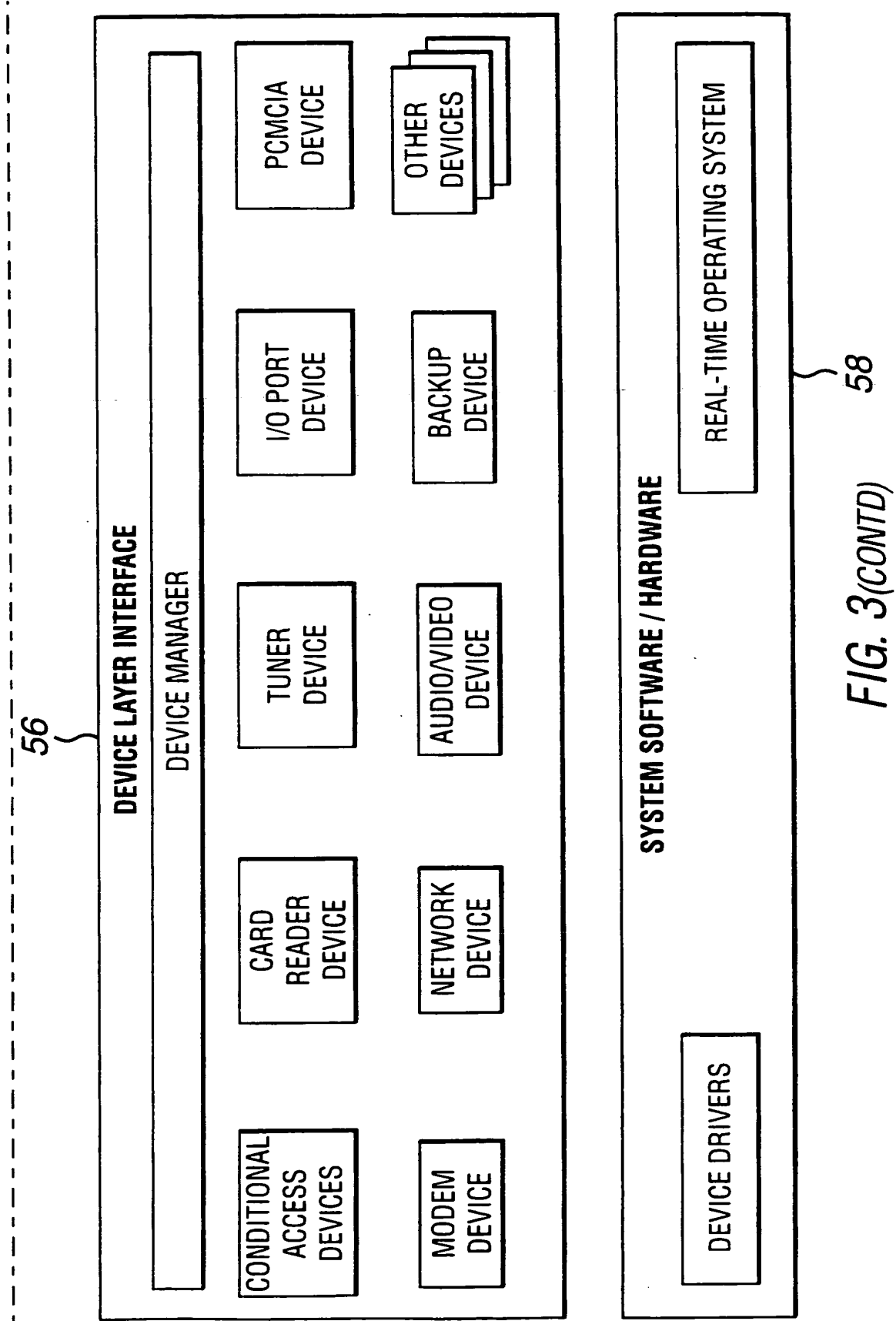


FIG. 3(Contd)

5/16
FIG. 4

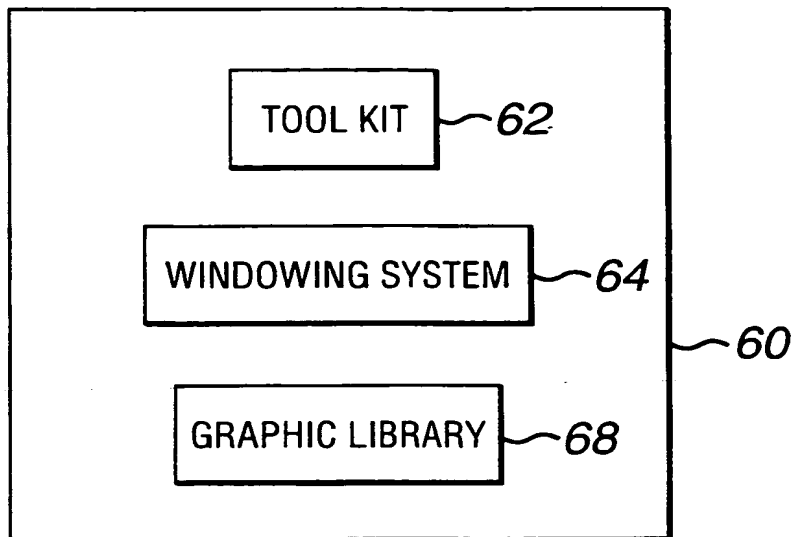
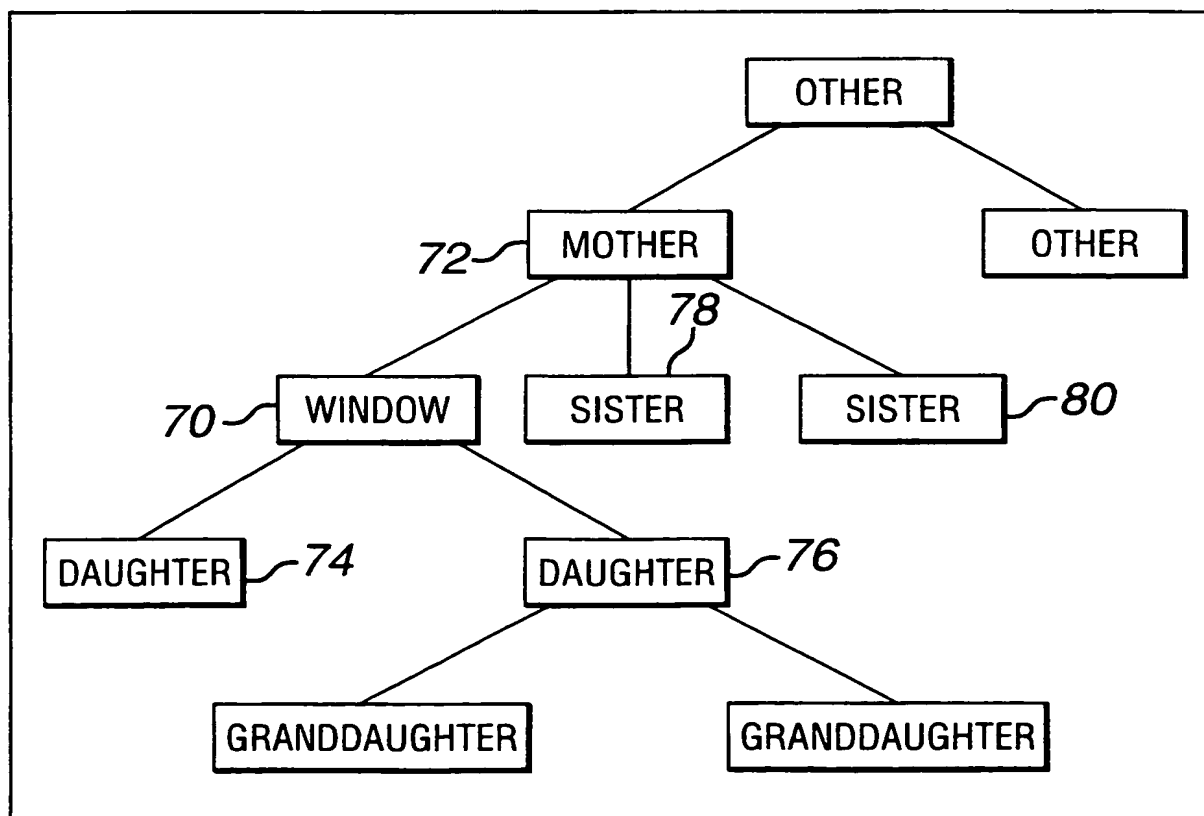


FIG. 5



6/16

FIG. 6(a)

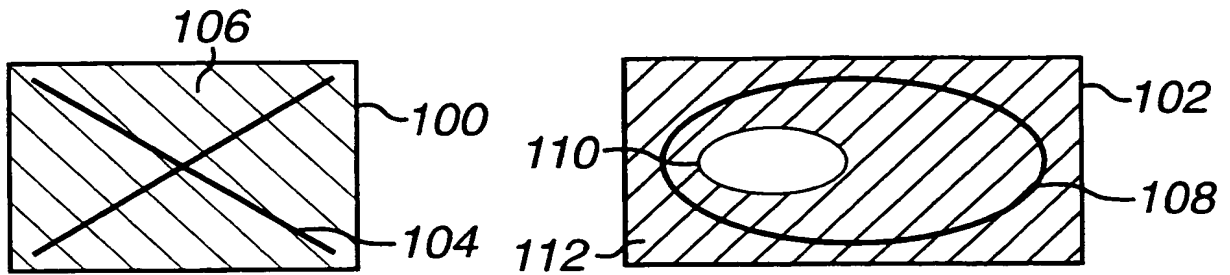


FIG. 6(b)

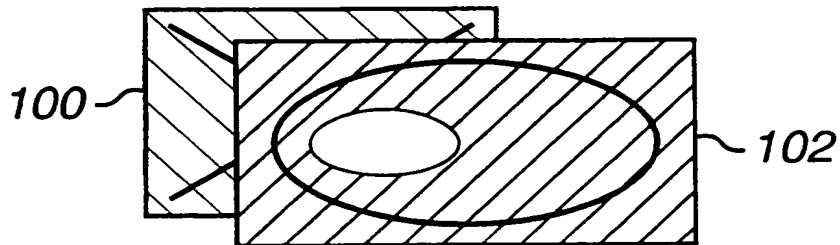
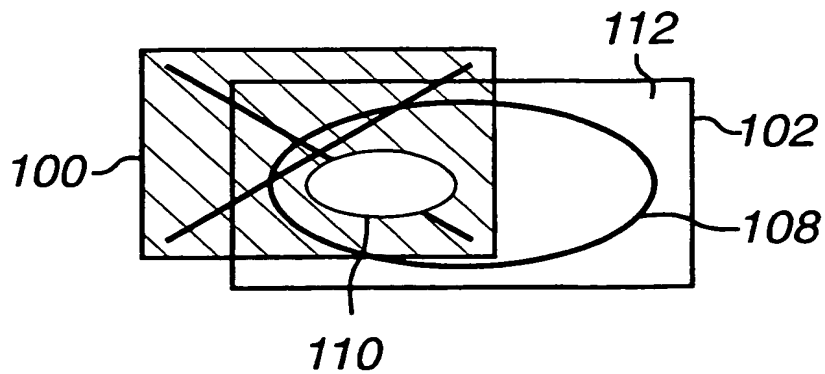


FIG. 6(c)



7/16

FIG. 7(a)

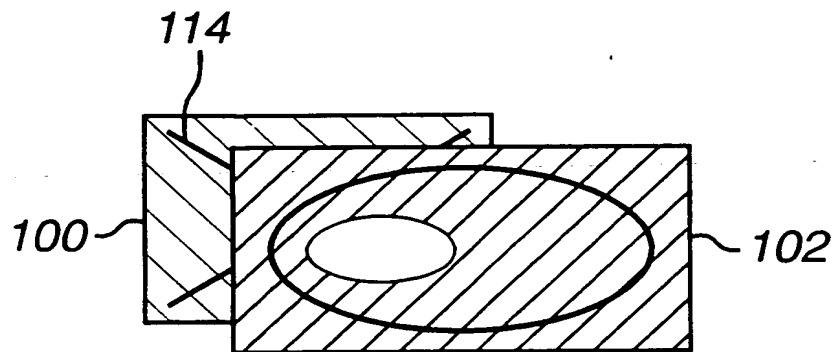
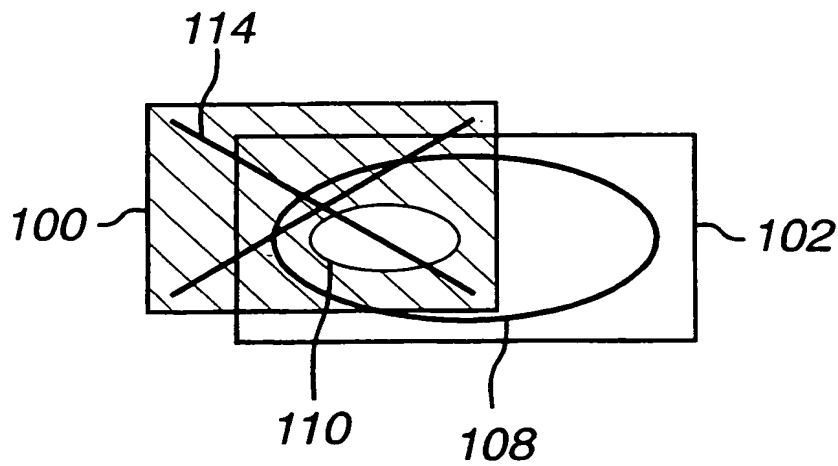


FIG. 7(b)



8/16

FIG. 8(a)

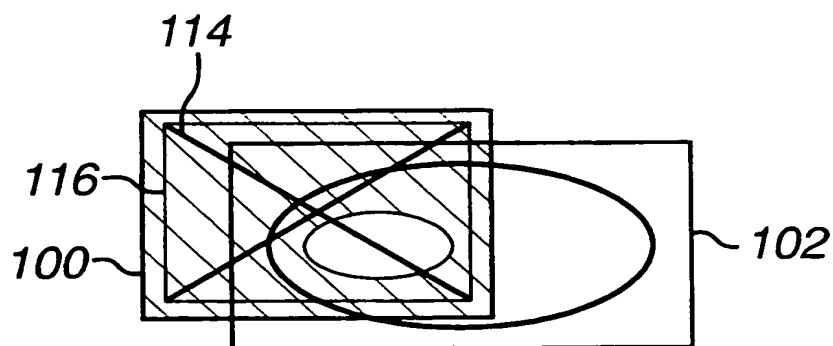


FIG. 8(b)

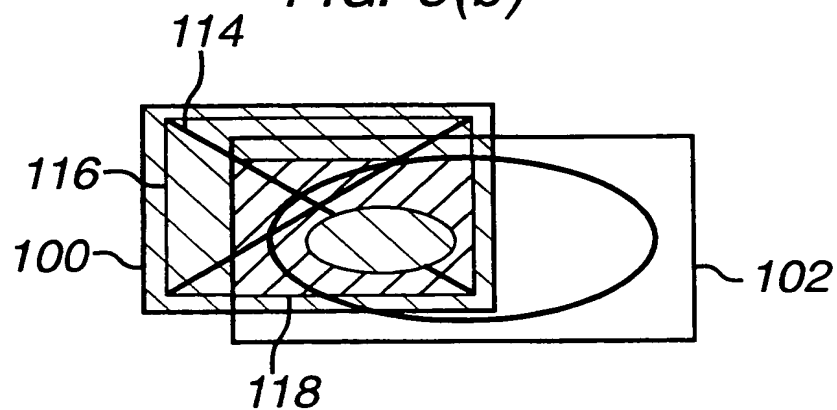
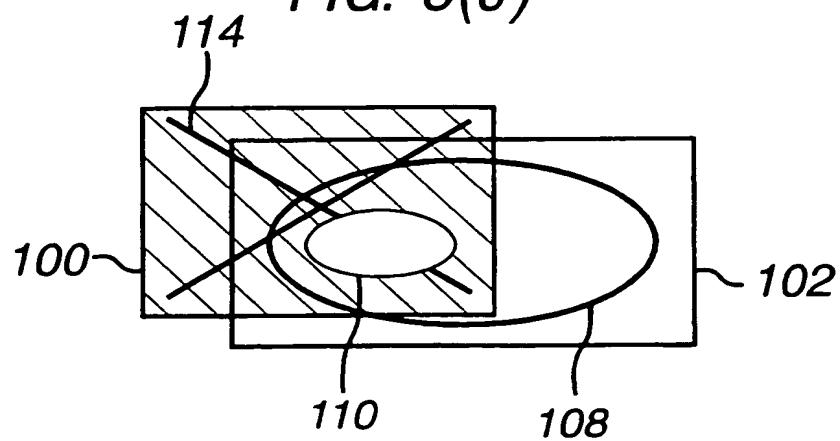


FIG. 8(c)



9/16

FIG. 9(a)

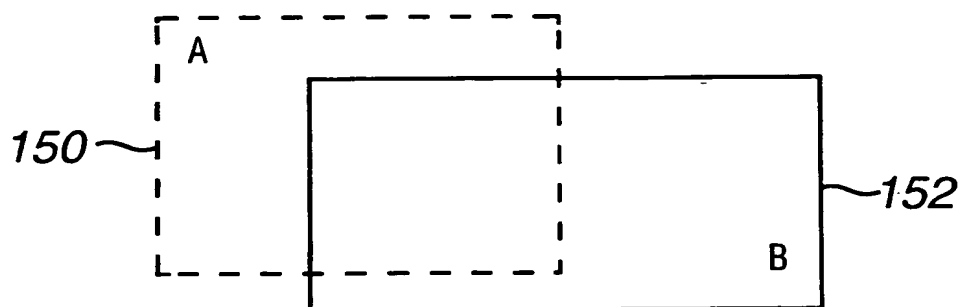


FIG. 9(b)

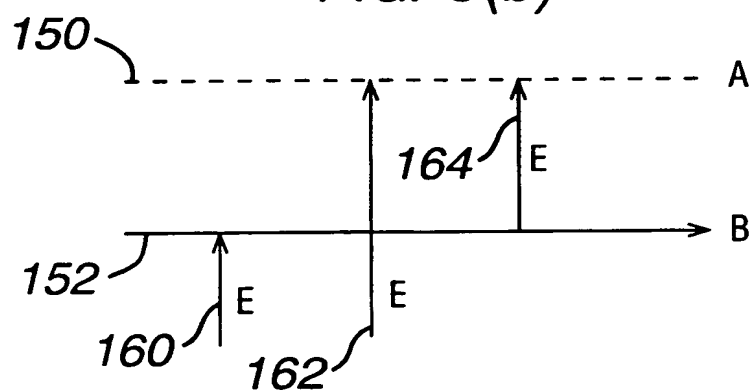
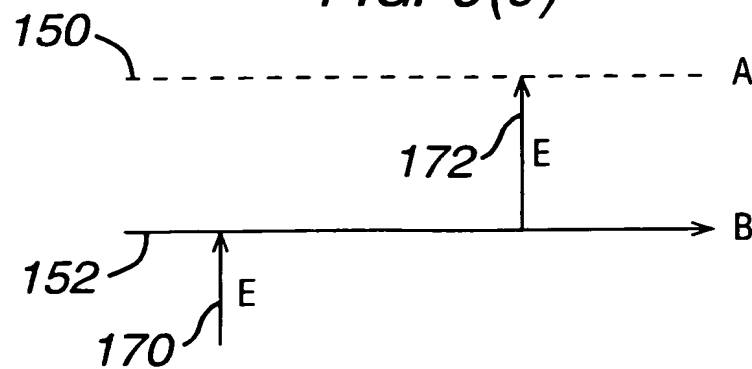
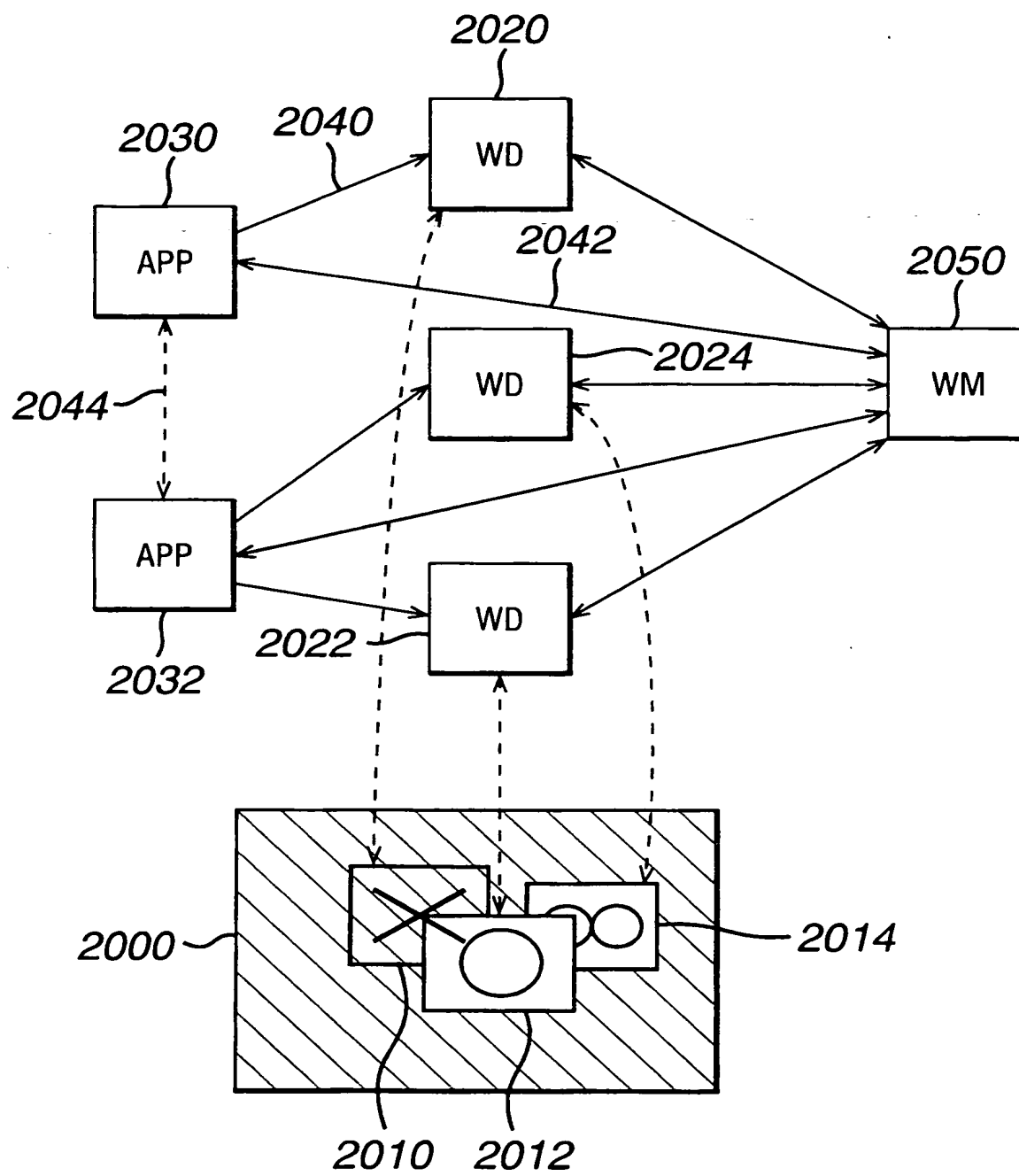


FIG. 9(c)



10/16

FIG. 10



11/16

FIG. 11(a)

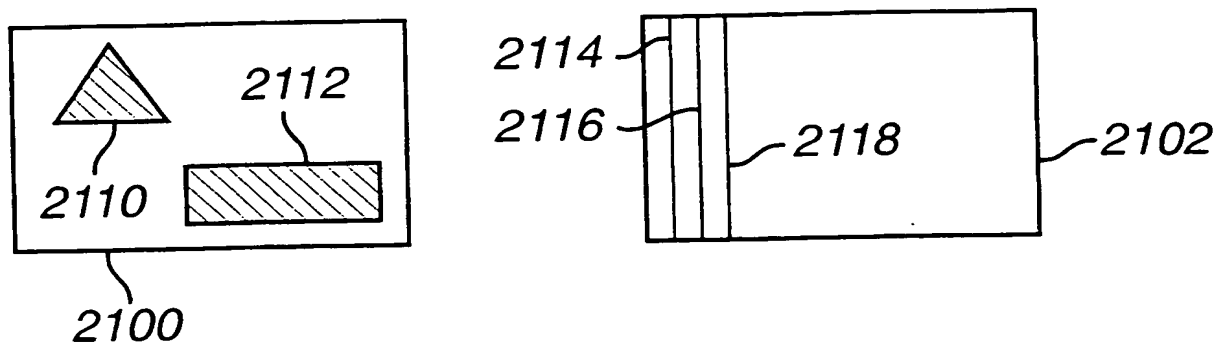


FIG. 11(b)

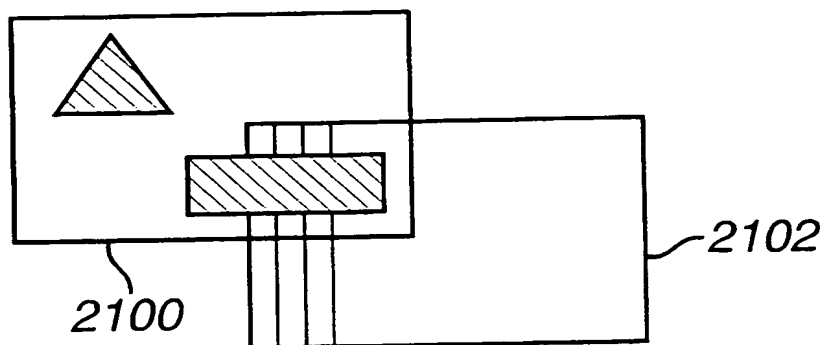
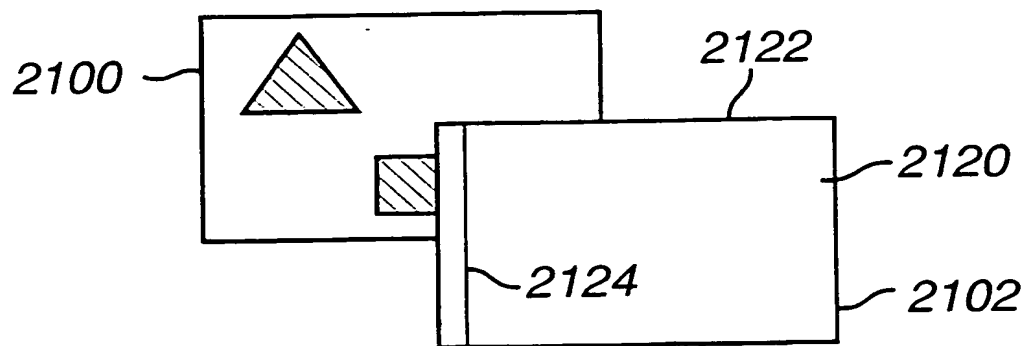


FIG. 11(c)



12/16

FIG. 11(d)

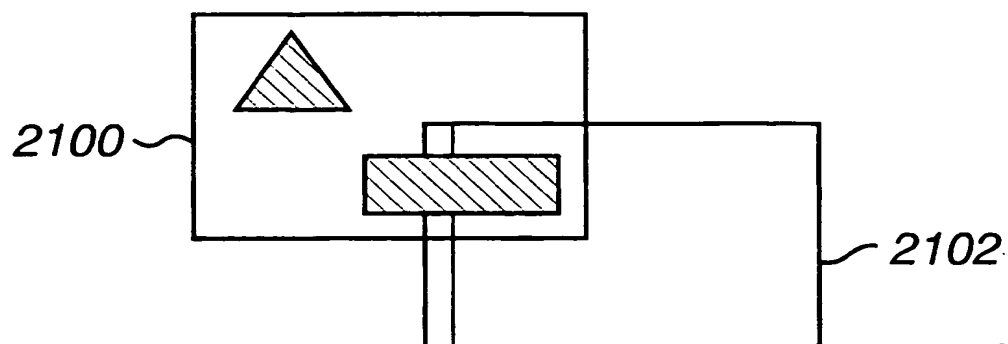


FIG. 11(e)

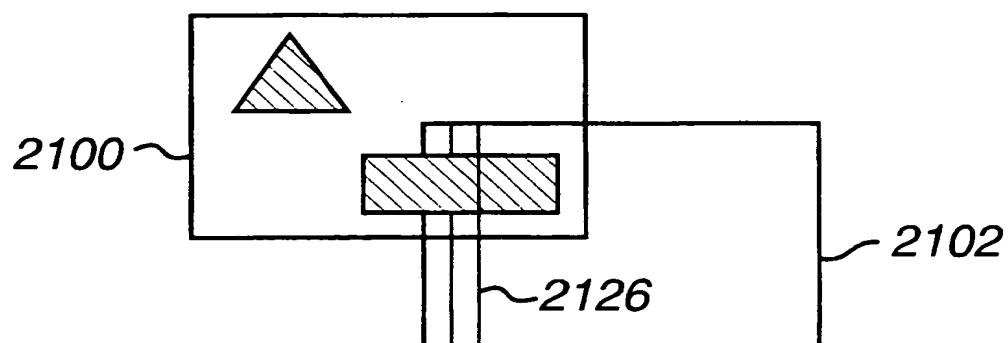
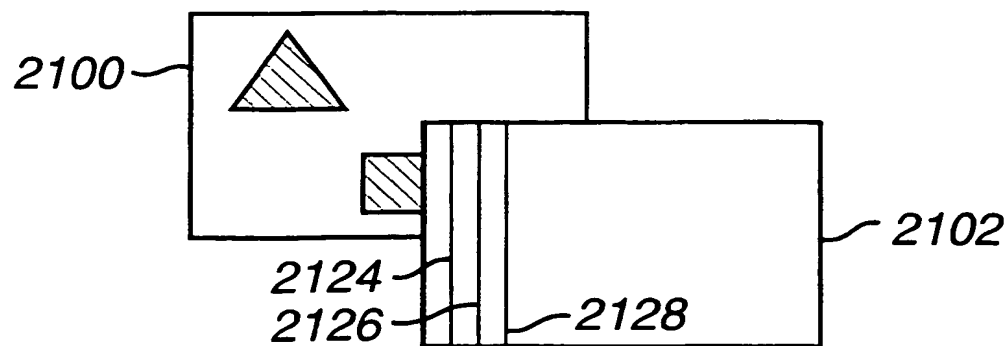
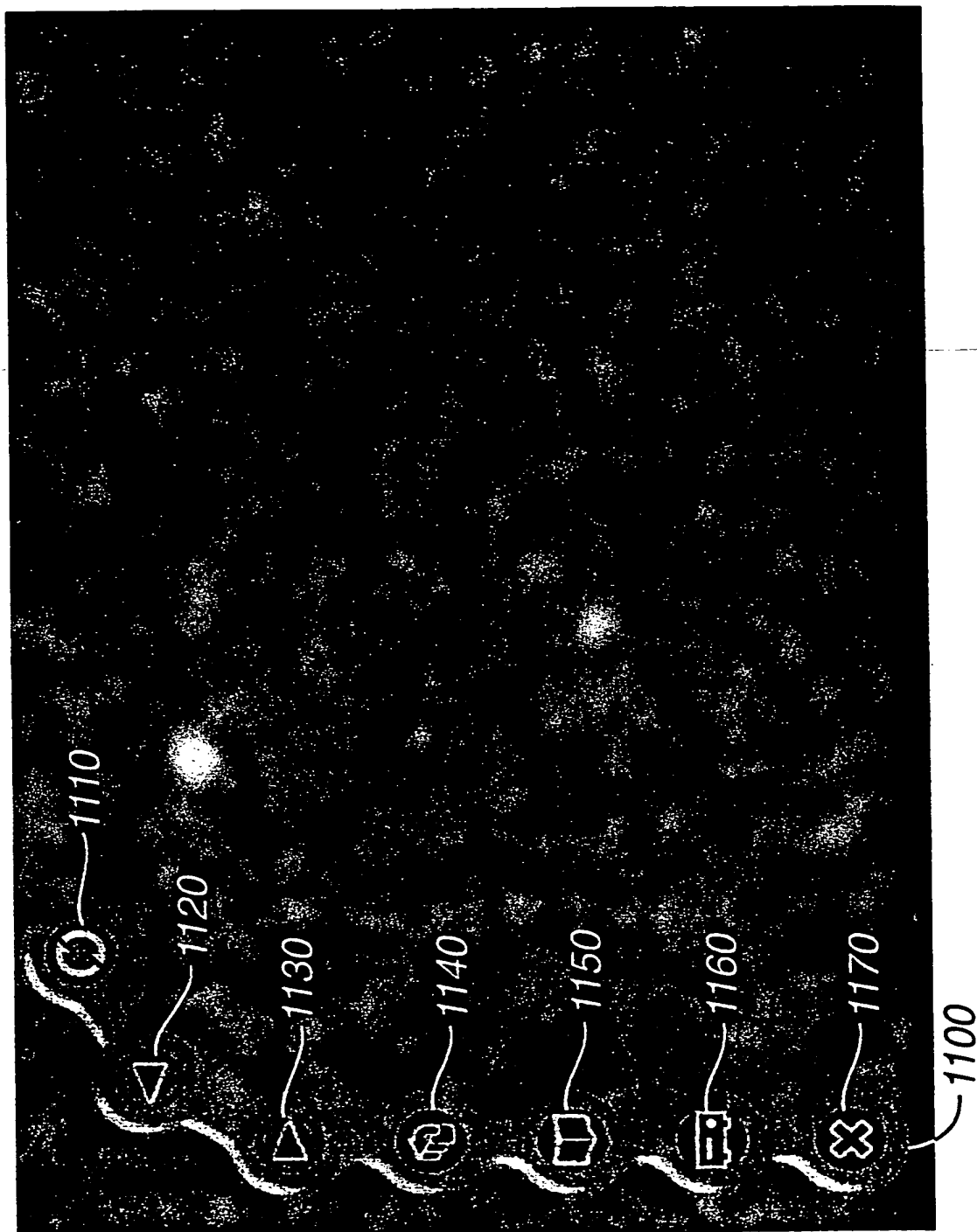


FIG. 11(f)



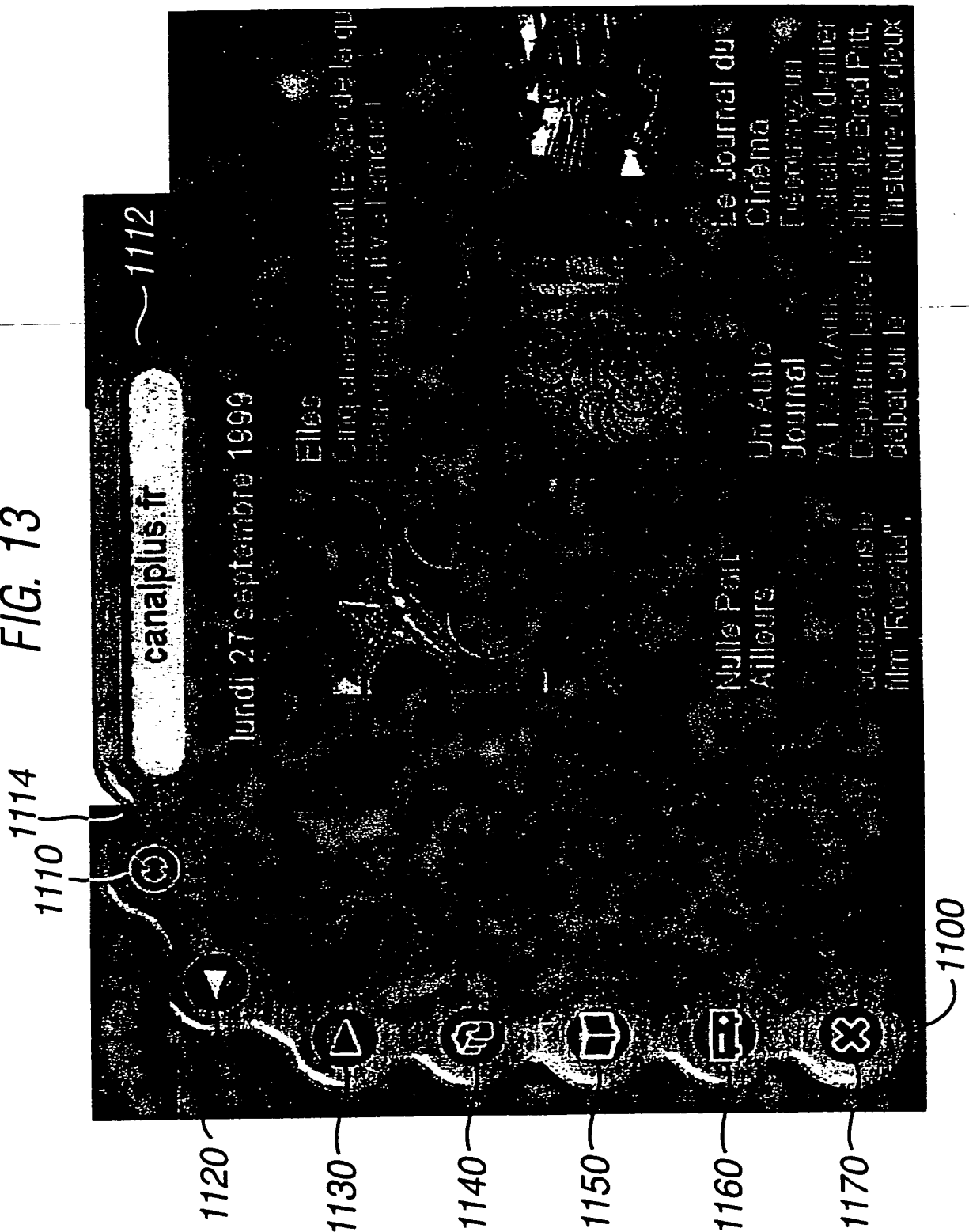
13/16

FIG. 12



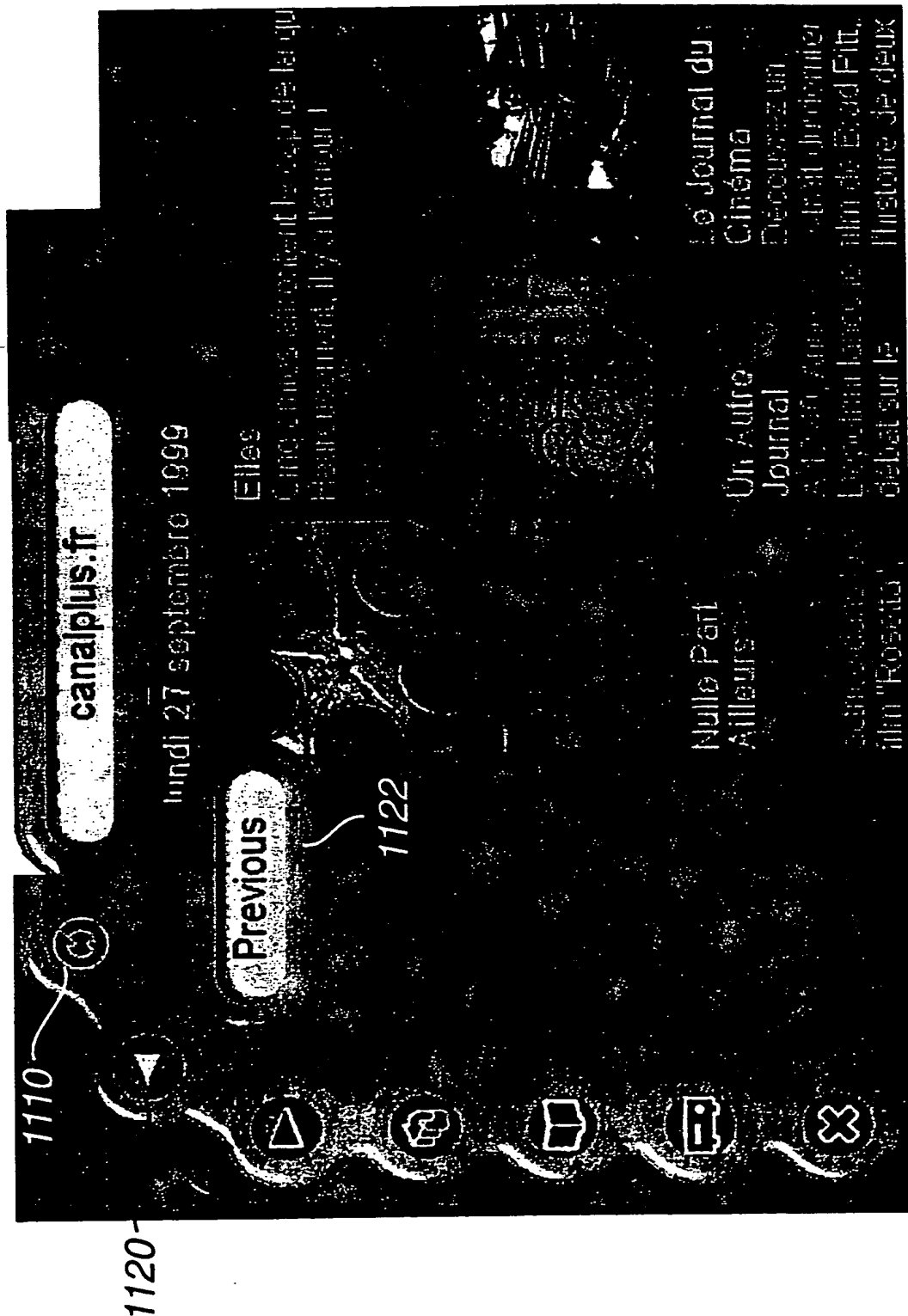
14/16

FIG. 13



15/16

FIG. 14



16/16

FIG. 15

